SRI SIDDHARTHA ACADEMY OF HIGHER EDUCATION

(*Declared as Deemed to be University Under Section 3 of the UGC Act, 1956*
Approved by AICTE, Accredited by NBA, NAAC 'A' Grade)
AGALKOTE, TUMAKURU – 572107
KARNATAKA

A Project Report
On
"INTRUSISHIELD : NAVIGATING SAFELY THROUGH CYBER TIDES"

Submitted in partial fulfillment of requirements for the award of degree

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

ABHIJEET BIRADAR	**(20CS002)**
BASAVARAJ SAJJAN	**(20CS016)**
A.JAYAKAR	**(20CS009)**
DARSHAN H	**(20CS025)**

Under the guidance of

Dr. CHANNAKRISHNARAJU
B.E,M.S., Ph.D., MISTE
Professor, Dept. of CS&E
SSIT, Tumakuru – 572105

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRI SIDDHARTHA INSTITUTE OF TECHNOLOGY

(A Constituent College of Sri Siddhartha Academy of Higher Education,
Approved by AICTE, Accredited by NBA, NAAC 'A' Grade)
MARALUR, TUMAKURU-572105

2023-2024

SRI SIDDHARTHA INSTITUTE OF TECHNOLOGY

(A Constituent College of Sri Siddhartha Academy of Higher Education, Tumakuru,
Approved by AICTE, accredited by NBA, NAAC 'A' Grade)
MARALUR, TUMAKURU – 572105, KARNATAKA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

*This is to certify that the project entitled **"IntrusiShield:Navigating Safely Through Cyber Tides"** is a bonafide work carried out by **ABHIJEET BIRADAR, BASAVARAJ SAJJAN, A.JAYAKAR & DARSHAN H** inpartial fulfillment for the award of degree of **Bachelor of Engineering** in **Computer Science and Engineering** during the academic year **2023-24**.*

It is certified that all the corrections/suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the degree of Bachelor of Engineering in Computer science and Engineering.

Signature of the guide
Dr.Channakrishnaraju
B.E., M.S., Ph.D., MISTE.,
Professor, Dept. of CSE
SSIT, Tumakuru

Signature of the HOD
Dr.Renukalatha S
B.E, M.S, Ph.D, MISTE.,
Dean(Academics) HOD, Dept. of CSE,
SSIT, Tumakuru

Signature of the Principal
Dr. M.S Raviprakash
B.E., M.Tech., Ph.D., FIE.,
Principal,
SSIT, Tumakuru

Project associates	**USN No.**
1) ABHIJEET BIRADAR	(20CS002)
2) BASAVARAJ SAJJAN	(20CS016)
3) A.JAYAKAR	(20CS009)
4) DARSHAN H	(20CS025)

External Examiners **Signature with Date**

1. ______________________________ ______________________________

2. ______________________________ ______________________________

ACKNOWLEDGEMENT

At the outset we express our most sincere grateful thanks to holy sanctum of **"SRI SIDDHARTHA INSTITUTE OF TECHNOLOGY"**, the temple of learning, for giving us an opportunity to pursue the degree course in Computer Science and Engineering thus help shaping our career.

We wish to express our sincere gratitude to **Dr. M. S. RAVIPRAKASHA** Principal, Sri Siddhartha Institute of Technology, for providing us an excellent academic environment, which has nurtured our practical skills, and for kindly obliging our requests and providing timely assistance.

We are highly grateful **Dr. RENUKALATHA S,** Dean academics and Head of Department of Computer Science and Engineering, for patronizing us throughout the project and for his encouragement and moral support.

We also wish to express our deep sense of gratitude to our project guide **Dr. CHANNAKRISHNARAJU,** Professor, Department of Computer Science and Engineering, for his valuable suggestions, guidance, moral support and encouragement in completion of this project successfully. We have been fortunate for having hisprecious help.

Finally, we express our gratitude to all the teaching and non-teaching of the **Computer Science and Engineering Department** staff for their timely support and suggestions.

We are also thankful to our parents for their moral support and constant guidance made our efforts fruitful. Last but not the least, our friends, without their constructive criticisms and suggestions we wouldn't have been able to complete successfully.

PROJECT ASSOCIATES

Abhijeet Biradar	(20CS002)
Basavaraj Sajjan	(20CS016)
A Jayakar	(20CS009)
Darshan H	(20CS025)

DECLARATION

We hereby declare that, this project work entitled ***"IntrusiShield:Navigating Safely Through Cyber Tides"*** is an original and bonafide work carried out by us at **SRI SIDDHARTHA INSTITUTE OF TECHNOLOGY,** Tumakuru, in partial fulfillment of **BACHELOR OF ENGINEERING** in **Computer Science and Engineering**.

We also declare that, to the best of our knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion by any student.

Date:

Place:

Abhijeet Biradar	(20CS002)
Basavaraj Sajjan	(20CS016)
A Jayakar	(20CS009)
Darshan H	(20CS025)

ABSTRACT

In the rapidly evolving landscape of cybersecurity, the necessity for robust Intrusion Detection Systems (IDS) to safeguard networks and sensitive information has become paramount. This project proposes an intelligent IDS that leverages advanced machine learning techniques for enhanced accuracy and efficiency. By combining traditional rule-based methods with state-of-the-art algorithms, the system is designed to detect and mitigate various types of network intrusions. A diverse dataset encompassing normal traffic and various attacks is used to train the system, enabling it to learn patterns indicative of normal and malicious behaviors. The system adapts dynamically to evolving threats by continuously updating its knowledge base, and employs anomaly detection algorithms to identify deviations from established patterns, facilitating timely detection of novel and sophisticated attacks.

To improve efficiency and reduce false positives, feature selection and dimensionality reduction techniques are incorporated. The proposed IDS includes real-time monitoring capabilities for swift response to emerging threats. Visualization tools are integrated to provide administrators with insightful representations of network activity and detected intrusions. This multi-layered approach aims to deliver a scalable, adaptive, and intelligent IDS capable of effectively responding to security threats in diverse network environments. Extensive testing in a controlled environment demonstrates the system's ability to protect against a wide range of cyber threats while minimizing false positives.

Additionally, the project developed an Intrusion Prevention System (IPS) to detect and prevent malicious files. Using different CNN models(VGG16, VGG19, Xecption, Resnet, Inception, Inception-Resnet), the IPS analyzes file attributes and behaviors, integrating signature-based detection with heuristic analysis to recognize both known and zero-day threats. The system continuously updates its threat database, ensuring robust protection. Feature selection and dimensionality reduction streamline the analysis process and reduce false positives, while real-time monitoring and automated response capabilities allow the IPS to swiftly quarantine or remove malicious files. Through rigorous testing, the IPS has proven its effectiveness in identifying and mitigating a wide array of malicious files, enhancing overall network security. This project demonstrates the potential of combining IDS and IPS technologies to create a comprehensive, adaptive, and intelligent security solution.

CONTENTS

LIST OF FIGURES

FIGURES **PAGE NO**

CHAPTER – 1

INTRODUCTION

In today's digital era, the proliferation of cyber threats has necessitated the development of robust cybersecurity measures. Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) are critical components of a comprehensive security strategy, designed to detect and mitigate malicious activities targeting networks and information systems. The increasing sophistication and volume of cyber-attacks, coupled with the growing complexity of network environments, have rendered traditional security mechanisms insufficient. This project addresses these challenges by developing an intelligent IDS and IPS leveraging advanced machine learning techniques.

The necessity for an intelligent IDS arises from the limitations of conventional systems. Traditional IDS rely heavily on rule-based methods, which are effective against known threats but often fail to detect novel and sophisticated attacks. Moreover, these systems tend to generate a high number of false positives, overwhelming security administrators and reducing operational efficiency. By integrating machine learning algorithms, the proposed IDS aims to enhance detection accuracy and adapt dynamically to evolving threats. This adaptive capability is crucial in maintaining robust security postures in the face of rapidly changing attack vectors.

To complement the IDS, the project also develops an IPS focused on detecting and preventing malicious files from compromising network security. The IPS employs machine learning techniques to analyze file attributes and behaviors, integrating signature-based detection with heuristic analysis. This dual approach enables the IPS to recognize both known malware and zero-day threats, providing a robust defense against a diverse array of malicious activities. The IPS is designed to update its threat database continuously, ensuring protection against emerging threats.

The project employs a systematic approach to develop and validate the IDS and IPS. The first step involves data collection and preprocessing, where a diverse dataset encompassing normal traffic and various types of attacks is gathered. This dataset is used to train and test the machine learning models. Feature selection and dimensionality reduction techniques are applied to enhance the efficiency of the models and reduce false positives. The system architecture is designed to facilitate real-time monitoring and swift response to emerging threat

Anomaly detection algorithms form the core of the IDS, enabling it to identify deviations from established patterns. These algorithms are trained on historical data to learn normal network behavior and detect anomalies indicative of intrusions. The IPS, on the other hand, focuses on file-based threats, employing machine learning models to analyze and classify files based on their attributes and behaviors. The integration of signature-based detection with heuristic analysis allows the IPS to identify both known and unknown threats effectively.

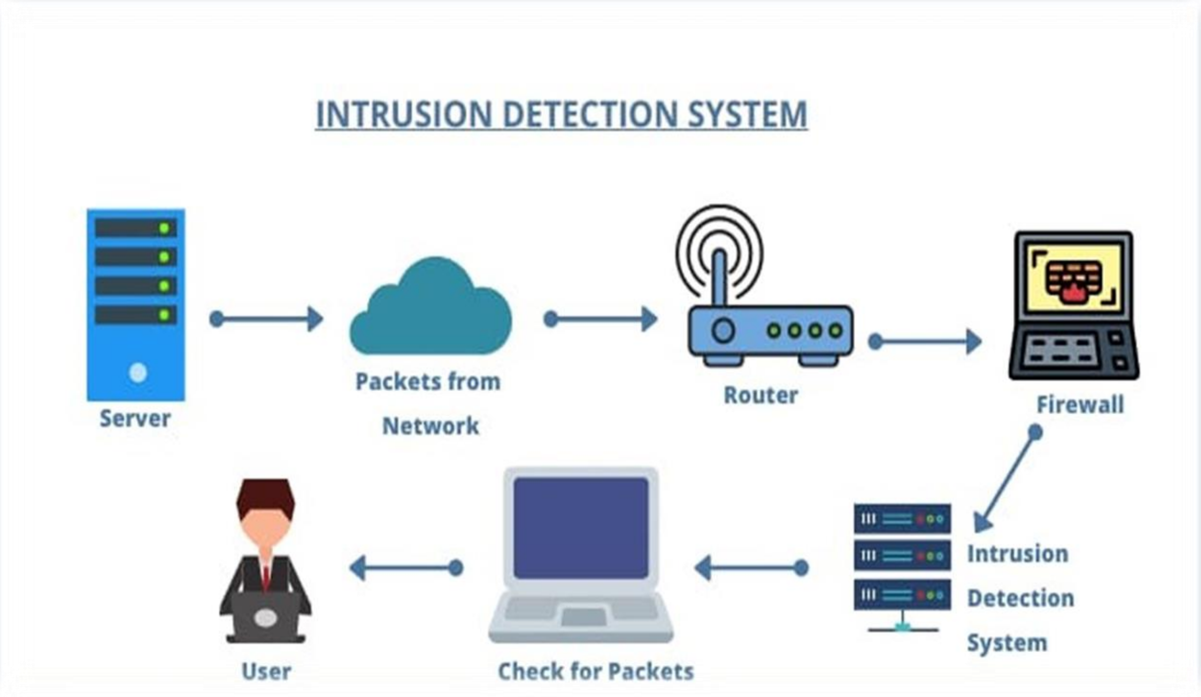


Fig. 1.1: Intrusion Detection System: An Overview

The diagram illustrates the flow of data in an Intrusion Detection System (IDS). It begins with a server sending packets through a network to a router. The router forwards these packets to a firewall, which filters the traffic and sends it to the IDS. The IDS monitors and analyzes the packets for any signs of malicious activity. The user, equipped with a computer, can check the packets being monitored by the IDS. This system ensures that potential threats are detected and addressed before they can affect the network or its components.

This project aims to contribute to the field of intrusion detection and prevention by developing a scalable, adaptive, and intelligent system capable of effectively identifying and responding to security threats in diverse network environments. The combination of IDS and IPS technologies, supported by advanced machine learning algorithms, offers a comprehensive solution to the challenges posed by modern cyber threats, paving the way for more secure and resilient information systems.

1.1. LITERATURE REVIEW

[1] Almutairi and Abdelmajeed (2017) present an innovative approach to enhancing the performance of signature-based Intrusion Detection Systems (IDS). Their study focuses on two primary advancements: the use of parallel processing and the minimization of the signature database. Traditional signature-based IDS rely heavily on comparing incoming traffic against a database of known signatures, which can be computationally intensive and time-consuming. The authors propose a parallel processing framework to distribute the computational load across multiple processors, significantly improving the system's speed and efficiency. Additionally, they introduce a method for minimizing the signature database by removing redundant or obsolete signatures without compromising detection accuracy. This reduction in database size not only speeds up the matching process but also reduces the overall system resource requirements. The experimental results presented in the conference paper demonstrate a marked improvement in both detection speed and system efficiency, highlighting the potential of parallel processing and database optimization in enhancing the performance of signature-based IDS.

[2] Walker-Roberts et al. (2018) delve into the emerging threats facing Cyber-Physical Systems (CPS), which integrate computing, networking, and physical processes. The study highlights the unique vulnerabilities of CPS, which arise from their interconnected nature and the critical functions they perform in sectors such as healthcare, transportation, and energy. They argue that traditional IT security measures are insufficient for CPS due to their real-time requirements and the potential for physical harm resulting from cyber attacks. The paper also discusses various security frameworks and methodologies tailored to CPS, underscoring the importance of integrating security into the design and operation phases of these systems. The insights provided by this research are crucial for understanding the complex security landscape of CPS and for developing robust defenses against both current and future threats.

[3] Yaacoub et al. (2020) explore the security challenges inherent in Cyber-Physical Systems (CPS), identifying key limitations and issues that need to be addressed to enhance their resilience. The paper outlines several critical vulnerabilities, including the lack of standardized security protocols, insufficient integration of security features, and the difficulty of ensuring real-time security in systems with stringent performance requirements. The authors also discuss the limitations of current security solutions, such as their inability to adapt to the dynamic and evolving nature of CPS environments. Future trends highlighted in the study include the development of more adaptive and intelligent security mechanisms, the integration of advanced technologies such as blockchain and artificial intelligence, and the need for comprehensive security testing frameworks. By addressing these challenges and exploring future directions, this paper provides valuable insights into the development of more robust and secure CPS.

[4] Ammar et al. (2015) propose the use of a decision tree classifier for tagging the priority of detected intrusions. Their approach aims to enhance the efficiency of IDS by not only detecting intrusions but also categorizing them based on their severity. This priority tagging helps in focusing the response efforts on the most critical threats, thereby optimizing resource allocation and incident response times. The decision tree classifier is trained on a dataset of network traffic, with features selected to maximize the accuracy of classification. The results of their experiments show that the decision tree classifier can effectively distinguish between different types of intrusions and assign appropriate priority levels. This method of prioritization is particularly beneficial in large-scale network environments where the volume of detected threats can be overwhelming. By implementing this approach, organizations can improve their incident response strategies and enhance overall network security.

1.2.PROBLEM STATEMENT

EXISTING PROBLEM:

In the dynamic landscape of cybersecurity, the effectiveness of traditional Intrusion Detection Systems (IDS) and Prevention Systems (IPS) is increasingly challenged by the rapid evolution of sophisticated cyber threats. Current systems often struggle with high false positive rates, limited adaptability to novel attack vectors, and inefficiencies in real-time threat detection and response. Moreover, the reliance on static rule-based approaches in IDS and signature-based methods in IPS poses limitations in accurately identifying and mitigating both known and unknown threats, including zero-day attacks

PROPOSED SOLUTION

This project proposes the development of an intelligent IDS and IPS framework that integrates advanced machine learning techniques to enhance accuracy, adaptability, and efficiency in detecting and mitigating network intrusions and malicious files. The IDS component utilizes a diverse dataset and anomaly detection algorithms to dynamically learn and adapt to normal and malicious network behaviors. Feature selection and dimensionality reduction techniques are employed to optimize performance and reduce false positives, while real-time monitoring capabilities ensure swift response to emerging threats.

1.2. MOTIVATION

- Cyber threats are evolving rapidly, posing challenges to current Intrusion Detection Systems (IDS) and Prevention Systems (IPS). These systems often struggle with high false alarms and are ineffective against new attack methods like zero-day exploits and polymorphic malware

- There is a critical need for IDS and IPS solutions that can adapt in real-time and accurately detect both known and unknown threats. Machine learning offers promising capabilities to analyze diverse data sets and identify subtle patterns indicative of malicious activities.

- Real-time monitoring and automated response capabilities are essential for swiftly detecting and mitigating emerging threats. These features not only reduce response times but also minimize potential damage from cyber incidents, ensuring continuous network security.

- By integrating advanced analytics and visualization tools, the project aims to provide administrators with actionable insights to manage and secure network environments effectively. This comprehensive approach seeks to strengthen overall cybersecurity defenses against the increasingly complex landscape of cyber attacks.

1.4. OBJECTIVES

- The primary objective of this project is to develop an intelligent, multi-layered IDS that combines traditional rule-based methods with advanced machine learning algorithms.

- The system is built to learn from historical data, identifying patterns indicative of normal and malicious behaviors. By continuously updating its knowledge base, the IDS can adapt to new attack patterns and enhance its detection capabilities over time.

- To complement the IDS, the project also develops an IPS focused on detecting and preventing malicious files from compromising network security.The IPS employs machine learning techniques to analyze file attributes and behaviors, integrating signature-based detection with heuristic analysis.

CHAPTER – 2

THEORY AND CONCEPT

2.1. SOFTWARE REQUIREMENTS

1.Python (version 3.x):

Python's cross-platform compatibility ensures seamless deployment of your IDS and IPS across different operating systems, whether on Linux servers or Windows environments. Its integration capabilities with other languages and frameworks enable easy incorporation of database connections, visualization tools, and real-time monitoring functionalities, essential for enhancing network security resilience and operational efficiency.

In conclusion, Python empowers your project with agility and scalability, enabling rapid prototyping, efficient algorithm implementation, and seamless integration of advanced machine learning techniques. Its vibrant ecosystem and active community support ensure continuous innovation and adaptability, making Python indispensable for developing cutting-edge cybersecurity solutions like your IDS and IPS.

2. Flask:

Flask, a lightweight and flexible web framework for Python, is utilized for building the web-based interface and APIs necessary for your IDS and IPS. Flask's simplicity and extensibility make it ideal for developing interactive dashboards, RESTful APIs, and real-time monitoring tools that administrators can use to manage and analyze network activities, detect intrusions, and assess system performance.

Furthermore, Flask's open-source nature and active community contribute to its flexibility and scalability. Developers can extend Flask's functionalities through plugins and third-party integrations, ensuring that your IDS and IPS framework evolves with the changing cybersecurity landscape. By adopting Flask, your project gains a powerful framework for creating responsive, data-driven applications that bridge machine learning insights with actionable cybersecurity strategies.

3. Matplotlib, Seaborn:

Matplotlib and Seaborn are essential data visualization libraries for Python, enabling administrators to generate insightful representations of network activities, detected intrusions, and system performance within your IDS and IPS.

Matplotlib offers a comprehensive suite of plotting tools for creating static, animated, and interactive visualizations. Its wide-ranging capabilities include line plots, scatter plots, histograms, and heatmaps, providing administrators with versatile tools to explore and communicate complex cybersecurity data effectively.

Seaborn builds upon Matplotlib's functionality with a higher-level interface for statistical plotting. It simplifies the creation of complex visualizations such as distribution plots, pair plots, and categorical plots, enhancing the ability to uncover patterns and trends in network traffic and security incidents.

Together, Matplotlib and Seaborn empower cybersecurity professionals to interpret data insights intuitively, facilitating informed decision-making and proactive security measures. Their integration with Python's data analysis tools and machine learning libraries ensures seamless visualization of anomaly detection results, network traffic patterns, and system performance metrics, supporting continuous monitoring and optimization of your IDS and IPS framework.

CHAPTER – 3

METHODOLOGY

3.1. BLOCK DIAGRAM

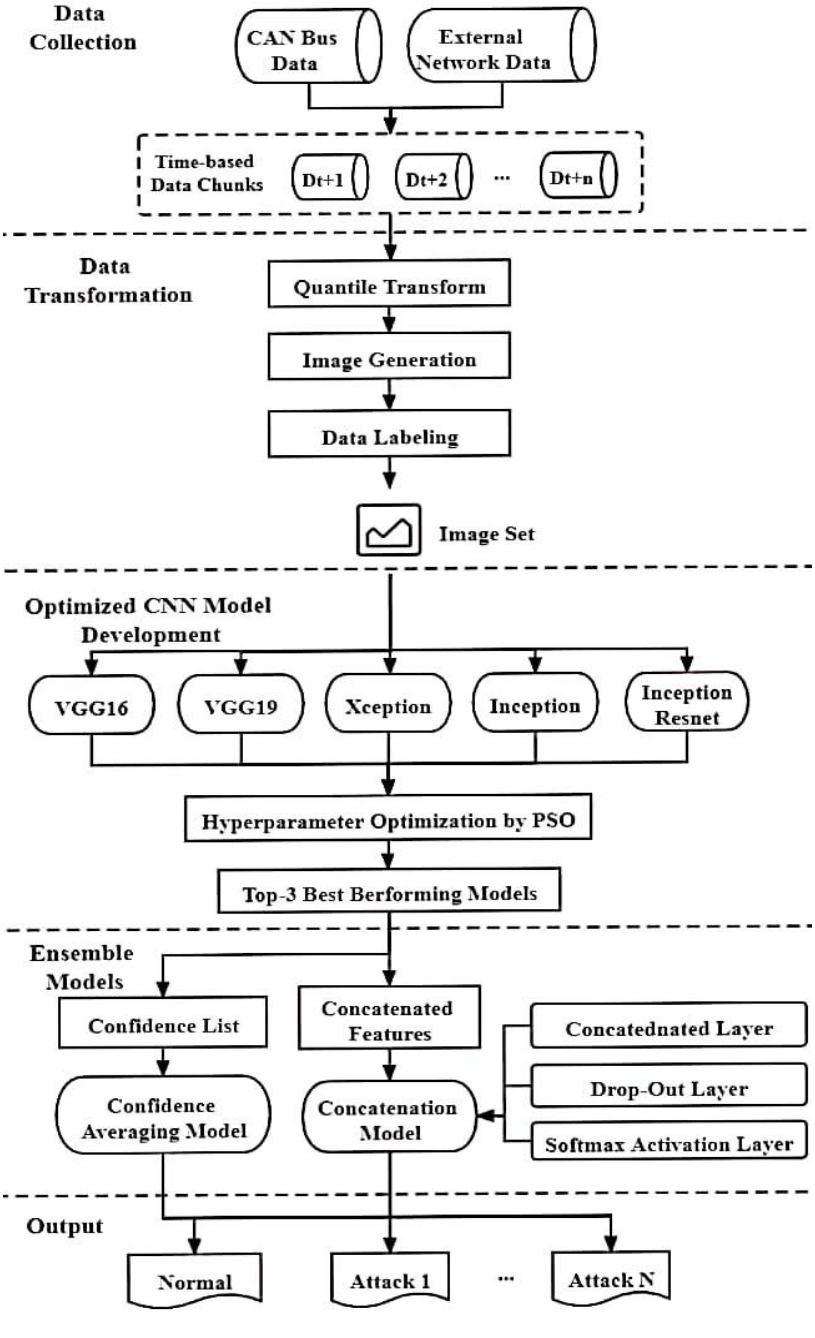


Fig. 3.1 : Block Diagram Illustrating an Intrusion Detection System

The diagram outlines a sophisticated framework designed to detect cyber attacks through the use of deep learning models. The process begins with data collection, where information from CAN Bus data (typical in vehicular systems) and external network data is gathered and segmented into time-based chunks for easier processing.

These chunks undergo a data transformation phase, where a quantile transformation normalizes the data, which is then converted into images and labeled accordingly. This transformed data forms the input for an optimized CNN model development phase. Here, multiple Convolutional Neural Network (CNN) architectures such as VGG16, VGG19, Xception, Inception, and Inception ResNet are trained. The performance of these models is enhanced through hyperparameter optimization using Particle Swarm Optimization (PSO), from which the top three performing models are selected.

In the ensemble models section, two distinct approaches are employed to combine the strengths of these top models. The first approach generates a confidence list and employs a confidence averaging model to aggregate the predictions. The second approach concatenates features from the models, feeding them into a concatenation model, which includes a concatenated layer, a dropout layer to prevent overfitting, and a softmax activation layer to produce probability distributions over the classes.

The final output provides a classification into various states such as normal operation or specific types of attacks. This workflow ensures a robust and accurate detection system capable of identifying and distinguishing between normal behavior and multiple forms of cyber attacks.

3.2 METHODOLOGY

The methodology involves a multi-step process combining data acquisition, transformation, model training, optimization, and ensemble learning to build an effective system for detecting and preventing cyber intrusions.

1. Data Collection:

The system collects data from two primary sources: the CAN Bus data from vehicle systems and external network data from various other sources.

2. Data Transformation:

- This step normalizes or scales the data, transforming it into a uniform distribution which can significantly improve model performance, especially for CNNs.
- The normalized data is converted into image formats. This is crucial as it allows leveraging CNN models, which are exceptionally good at processing image data.
- Each image is labeled to indicate whether it represents normal behavior or a specific type of attack. These labels are essential for training the CNN models in a supervised manner.

3. Optimized CNN Model Development:

CNN Architectures: Multiple pre-trained CNN architectures are used, including

- **VGG16 and VGG19:** Known for their deep architecture with sequential layers.
- **Xception:** An extension of the Inception architecture with depthwise separable convolutions.
- **Inception and Inception ResNet:** These architectures use inception modules that efficiently capture multi-scale features.
- **Hyperparameter Optimization using PSO:** Particle Swarm Optimization (PSO) is used to fine-tune the hyperparameters of each CNN to ensure the models perform optimally. PSO simulates the social behavior of birds or fish and is effective in finding optimal solutions in a multi-dimensional space.

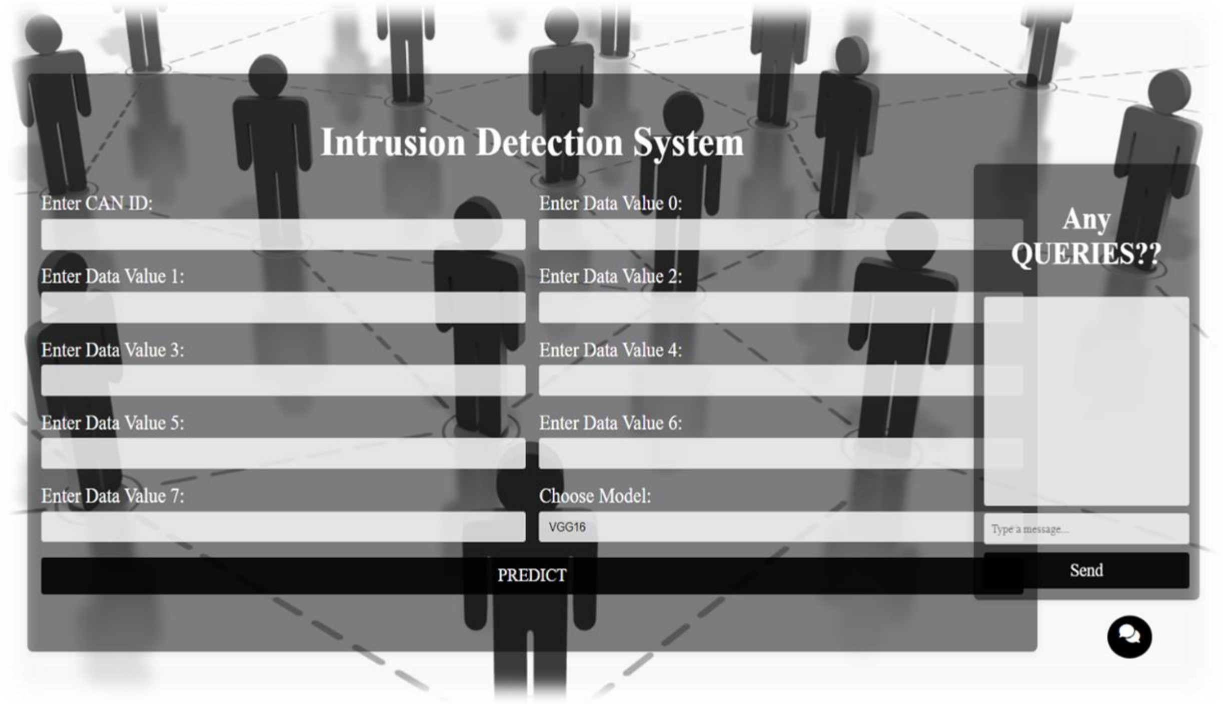
4. Output:

Classifications: The system outputs the classification of the input as either "Normal" or one of several attack types (Attack 1, Attack 2, ..., Attack N).

CHAPTER – 4

DESIGN

4.1. IDS DESIGN



Choose Model: VGG16

Fig. 4.1 : UI Design for Intrusion Detection System

CAN ID:

The CAN ID is a unique identifier that is part of every CAN frame (or message) transmitted over the network.

DATA VALUES[0-7]:

- In a CAN message , There are usually eight bytes of data, numbered 0 to 7,each containing information relevant to vehicle operation, such as sensor readings, control commands, or diagnostic data.
- Data values[0-7] typically represents the payload of a CAN message. Payload values are used to built traffic in the network.To create something abnormal in network.

CNN MODELS:

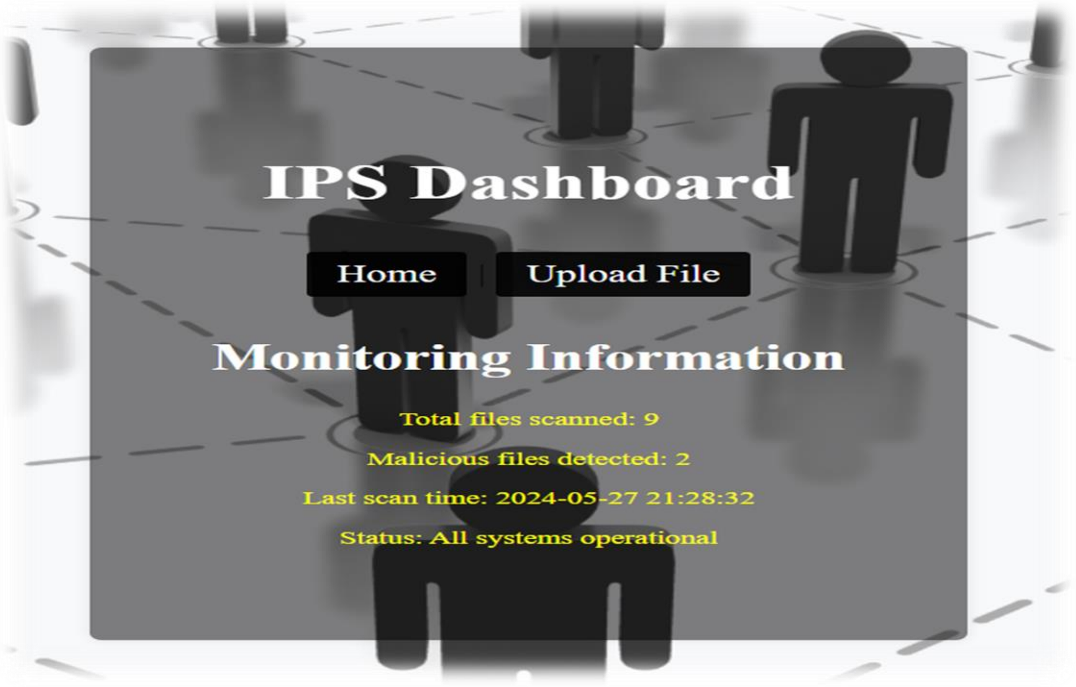
- VVG16
- VGG19
- XCEPTION

- RESNET
- INCEPTION
- INCEPTION RESNET

CHAT BOT:

Created a chat bot that clears the queries about the Intrusion detection system.

4.2. IPS DESIGN



IPS Dashboard

Monitoring Information

Total files scanned: 9

Malicious files detected: 2

Last scan time: 2024-05-27 21:28:32

Status: All systems operational

Fig.4.2 : UI Design for Intrusion Prevention System

- Two files are required
    1. Corrupt file
    2. Clean file
- When the corrupted file is uploaded in the IPS it detect the file as malicious
- When the clean file is uploaded in the IPS it detects the file as clean

4.3. CNN Models:

4.3.1. VGG16(visually geometry group) Model:

VGG16 is a deep convolutional neural network architecture. It consists of 16 layers with weights, including 13 convolutional layers and 3 fully connected layers, and is known for its simplicity and effectiveness in image classification tasks. Implementing VGG16 involves stacking multiple convolutional blocks followed by fully connected layers, and it typically achieves good performance on various image recognition benchmarks. Adjustments such as batch normalization and dropout can be added to improve generalization and training stability as needed.

VGG16 ALGORITHM:

1. **Input**:

   RGB Image of size 224×224×3224 \times 224 \times 3224×224×3.

2. **Convolutional Blocks**:

   - Sequentially apply convolutional layers with 3×33 \times 33×3 filters and ReLU activations.
   - Optionally apply batch normalization after each convolutional layer.
   - Apply max pooling after each convolutional block to reduce spatial dimensions.

3. **Fully Connected Layers**:

   - Flatten the output from the last convolutional block.
   - Connect to three fully connected layers with ReLU activations (except for the last layer).
   - Use softmax activation in the last fully connected layer for classification.

VGG16 is renowned for its simplicity and effectiveness in image classification, making it applicable in IDS projects where network traffic data can be represented as spectrograms or other image formats. By training VGG16 on a dataset of network packet features extracted in real-time, it can classify incoming traffic as normal or malicious based on learned patterns. Its deep architecture enables it to capture hierarchical features in data, potentially improving the IDS's ability to detect complex intrusion attempts.

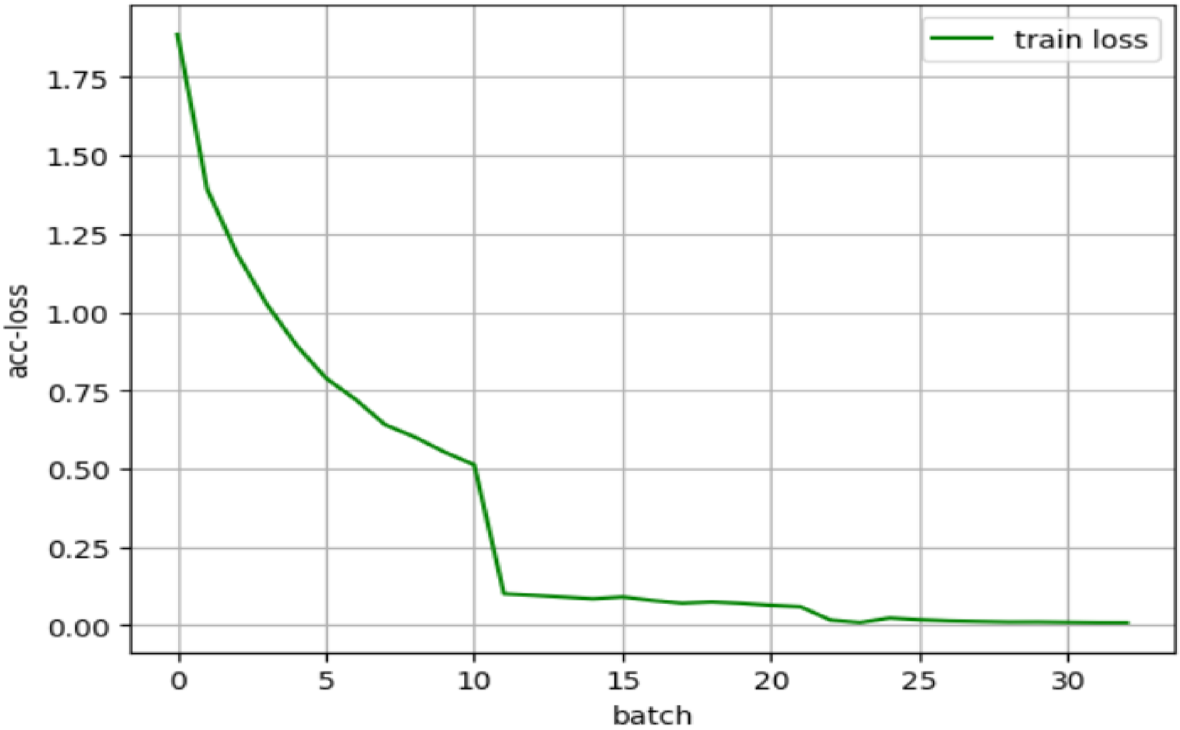


Fig. 4.3 : Training Accuracy-Loss Over Batches for VGG16 Model

Acc-Loss vs Batch graph

Training Accuracy : 100%

Prediction Accuracy : 74.44%

4.3.2. VGG19(visually geometry group) Model:

Extension of VGG16 with deeper architecture. VGG19 is a deep neural network architecture designed for image classification tasks. It is known for its simplicity and effectiveness in image recognition tasks. VGG19 is known for its simplicity and effectiveness in image classification tasks due to its deep architecture and small filter sizes. Implementing VGG19 involves stacking convolutional blocks followed by fully connected layers, and it typically performs well on various image recognition benchmarks with appropriate adjustments for specific tasks.

VGG19 ALGORITHM:

1. **Input**:

   RGB Image of size 224×224×3224 \times 224 \times 3224×224×3.

2. **Convolutional Blocks**:

   - Sequentially apply convolutional layers with 3×33 \times 33×3 filters and ReLU activations.
   - Optionally apply batch normalization after each convolutional layer.
   - Apply max pooling after each convolutional block to reduce spatial dimensions.

3. **Fully Connected Layers**:

   - Flatten the output from the last convolutional block.
   - Connect to three fully connected layers with ReLU activations (except for the last layer).
   - Use softmax activation in the last fully connected layer for classification.

VGG19, an extension of the VGG16 architecture, is well-suited for IDS applications where network traffic data is processed as image-like inputs, such as spectrograms or feature maps. Its deep architecture with multiple convolutional layers facilitates the extraction of intricate patterns and features from network traffic data. By training VGG19 on labeled datasets of network behaviors, it can effectively classify incoming traffic as normal or malicious based on learned representations. VGG19's sequential structure and uniform architecture make it particularly effective in detecting common intrusion signatures and anomalies in network communications. Its ability to capture hierarchical features through deep layers enhances the IDS's capability to discern complex attack patterns and anomalies with high accuracy, thereby strengthening the overall security posture by promptly identifying and mitigating network threats.

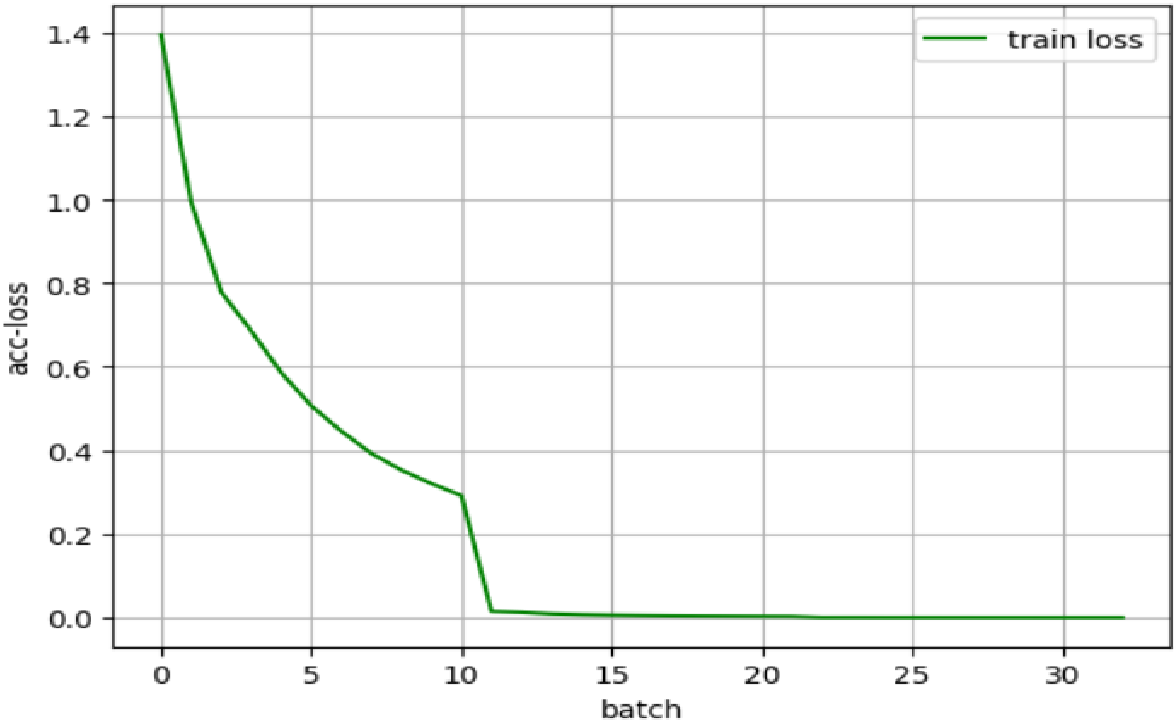


Fig. 4.4 : Training Accuracy-Loss Over Batches for VGG19 Model

<u>Acc-Loss vs Batch graph</u>

Training Accuracy : 99.84%

Prediction Accuracy : 74.34%

4.3.3. XCEPTION Model:

Xception is a deep learning-based approach for image classification tasks. It is a type of convolutional neural network (CNN) that uses a combination of convolutional and pooling layers to extract features from images. Xception is designed to balance between computational efficiency and model performance by leveraging depthwise separable convolutions. Implementing Xception typically involves configuring the number and size of layers based on the specific application requirements, but the fundamental structure revolves around depthwise separable convolutions and residual connections.

XECEPTION ALGORITHM:

1. **Input**:

   RGB Image of size H×W×3H \times W \times 3H×W×3.

2. **Initial Stage**:

   Apply a standard convolutional layer to the input image.

3. **Middle Flow**:

   - Repeat a series of modules, each containing:
     - Depthwise separable convolutions (optionally followed by batch normalization and ReLU).
     - Residual connection (skip connection).

4. **Exit Flow**:

   - Apply a series of depthwise separable convolutions.
   - Use pooling (e.g., global average pooling) to reduce spatial dimensions.
   - Flatten the output.
   - Apply fully connected layers for final classification or regression.

Xception, an extension of convolutional neural networks (CNNs), excels in feature extraction from complex images. In the context of IDS, Xception can be employed for analyzing network traffic patterns captured as image-like data. By training on labeled datasets of normal and anomalous network behaviors, Xception can automatically learn and distinguish between benign and malicious network activities. Its ability to capture intricate dependencies in data makes it suitable for detecting subtle anomalies and enhancing the accuracy of intrusion detection systems.

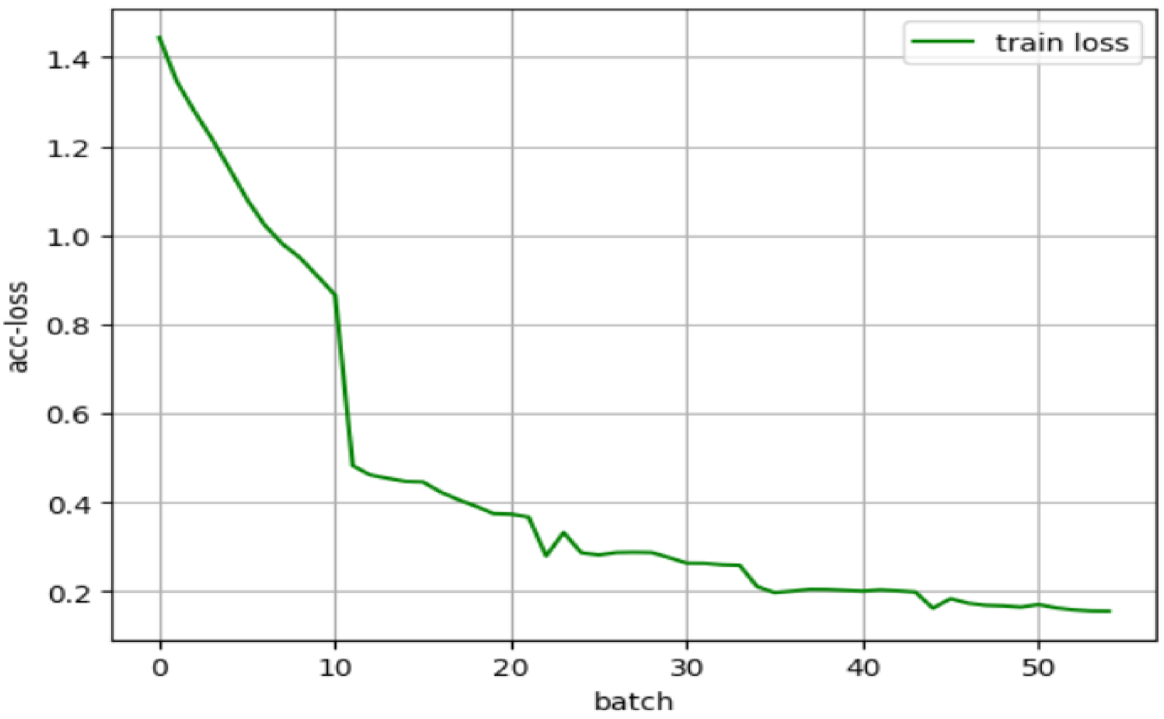


Fig. 4.5 : Training Accuracy-Loss Over Batches for Xception Model

Acc-Loss vs Batch graph

Training Accuracy : 98.80%

Prediction Accuracy : 77.14%

4.3.4. RESNET(Residual Network) Model:

It is Deep CNN Model known for its innovative residual connection which enables training of very deep networks. It removes the vanishing gradient problem. Implementing ResNet involves defining residual blocks and arranging them sequentially, culminating in pooling and fully connected layers for classification. Adjustments such as batch normalization and dropout can be incorporated to enhance model performance and training stability as needed.

RESNET ALGORITHM:

1. **Input**:

   RGB Image of size 224×224×3224 \times 224 \times 3224×224×3.

2. **Initial Convolutional Layer**:

   Apply a standard convolutional layer to the input image.

3. **Residual Blocks**:

   Sequentially apply residual blocks, each containing two convolutional layers with skip connections.

4. **Pooling and Fully Connected Layers**:

   - Apply average pooling to reduce spatial dimensions.
   - Flatten the output and connect to fully connected layers with ReLU activation.
   - Use softmax activation in the last fully connected layer for classification.

ResNet's deep architecture with residual connections is particularly beneficial in IDS projects where deep learning models are required to process intricate network traffic data. By employing ResNet, IDS can effectively learn to classify network packets or flows as normal or anomalous based on learned patterns and anomalies. Its ability to mitigate the vanishing gradient problem and train very deep networks ensures robust performance in detecting complex and evolving cyber threats, making it a suitable choice for enhancing the detection capabilities of intrusion detection systems.

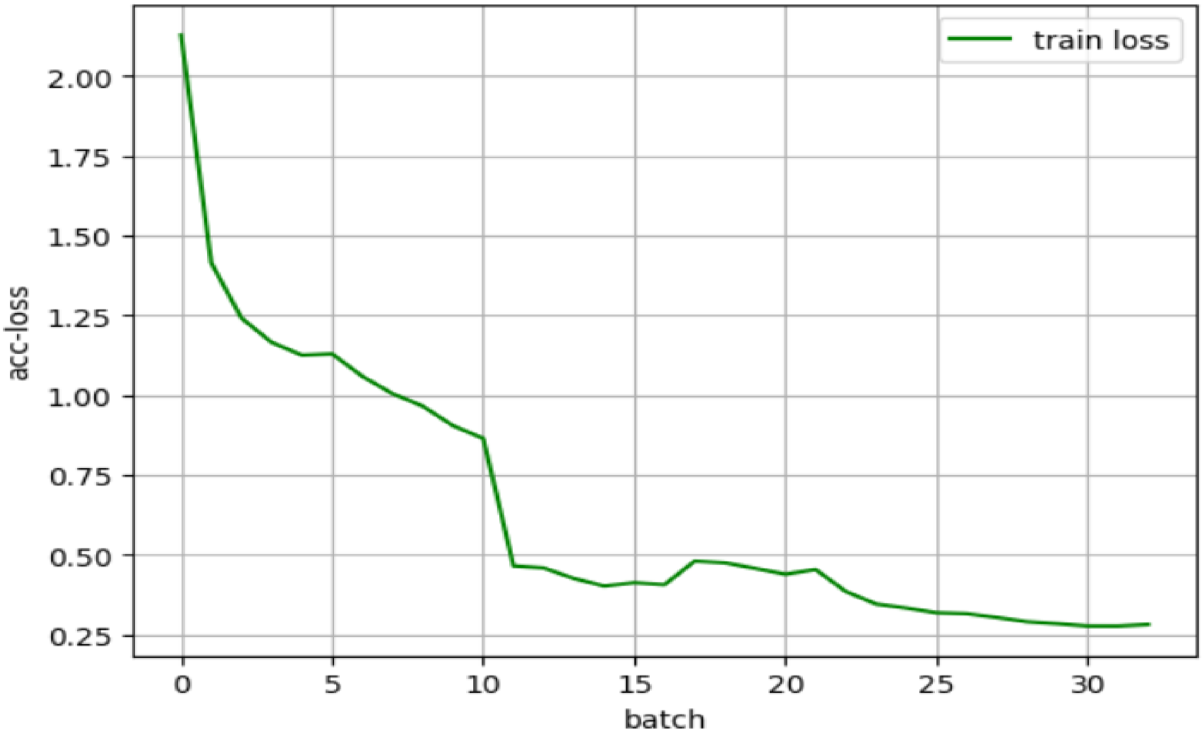


Fig. 4.6 : Training Accuracy-Loss Over Batches for ResNet Model

<u>Acc-Loss vs Batch graph</u>

Training Accuracy : 96.88%

Prediction Accuracy : 73.97%

4.3.5. INCEPTION Model:

Xception is a deep learning-based approach for image classification tasks. It is a type of convolutional neural network (CNN) that uses a combination of convolutional and pooling layers to extract features from images. Inception is designed to efficiently capture spatial hierarchies and patterns at different scales using Inception modules and reduction blocks. Implementing Inception involves configuring these modules and connecting them with appropriate pooling and convolutional operations. Adjustments such as dropout and batch normalization can further enhance the model's performance and generalization capabilities.

INCEPTION ALGORITHM:

1. **Input**:

   RGB Image of size H×W×3.

2. **Initial Convolution**:

   Apply a standard convolutional layer to the input image.

3. **Inception Modules**:

   Sequentially apply Inception modules with different filter sizes:

   - Each Inception module contains parallel convolutions (1x1, 3x3, 5x5, and optionally max pooling).
   - Concatenate the outputs of these operations along the channel dimension.

4. **Reduction Blocks**:

   Apply reduction blocks between Inception modules to reduce spatial dimensions using pooling and convolutions.

5. **Fully Connected Layers**:

   - Flatten the output from the last Inception module.
   - Connect to fully connected layers with dropout for regularization.
   - Use softmax activation in the last fully connected layer for classification.

Inception, known for its efficient multi-scale feature extraction capabilities, can be effectively utilized in an IDS project to analyze network traffic data. By treating network traffic data as image-like inputs, Inception can capture intricate patterns and dependencies across different scales. This architecture's ability to employ parallel convolutional operations of varying sizes enables it to detect subtle anomalies in network behaviors, such as unusual traffic patterns or protocol deviations. Inception's deep layers can learn hierarchical representations of network features, making it suitable for enhancing the accuracy and robustness of intrusion detection systems by automatically identifying both known and novel intrusion attempts with high precision.

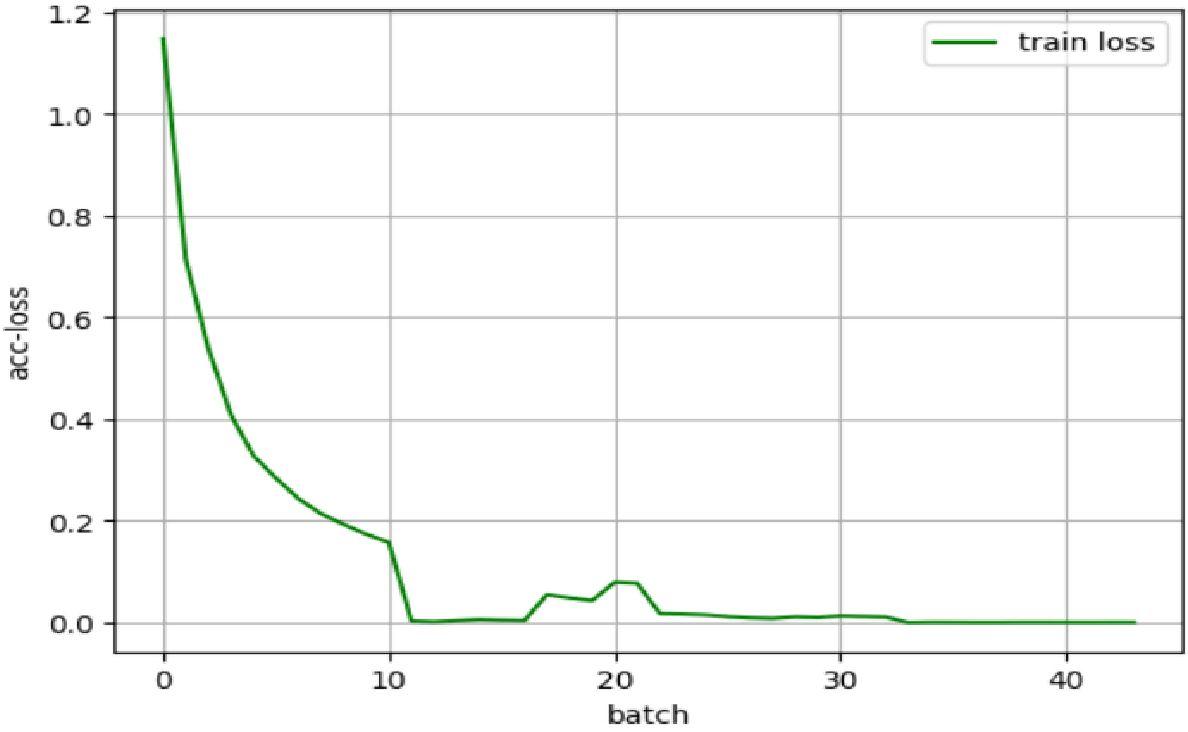


Fig. 4.7 : Training Accuracy-Loss Over Batches for Inception Model

Acc-Loss vs Batch graph

Training Accuracy : 98.86%

Prediction Accuracy : 86.00%

4.3.6. INCEPTION-RESNET Model:

Inception-ResNet is a variant of the Inception architecture that incorporates residual connections to improve the efficiency and performance of the network. This allows the network to learn more complex and deeper representations while reducing the risk of vanishing gradients. Inception-ResNet has been shown to achieve high performance on various image classification tasks. Inception-ResNet combines the strengths of both Inception and ResNet architectures, making it capable of handling complex image recognition tasks effectively. Implementing Inception-ResNet involves defining and organizing Inception-ResNet blocks with residual connections, followed by pooling and fully connected layers for classification or regression tasks. Adjustments such as batch normalization and dropout can be incorporated as needed to improve model performance and stability during training.

INCEPTION-RESNET ALGORITHM:

1. **Input**:

   RGB Image of size 299×299×3299 \times 299 \times 3299×299×3.

2. **Stem Block**:

   Initial convolutional layers and max pooling to process the input image.

3. **Inception-ResNet Modules**:

   Sequentially apply Inception-ResNet modules combining Inception blocks with residual connections.

4. **Reduction Blocks**:

   Apply reduction blocks between Inception-ResNet modules to reduce spatial dimensions using pooling and convolutions.

5. **Global Average Pooling and Fully Connected Layers**:

   - Apply global average pooling to reduce spatial dimensions to 1x1.
   - Flatten the output and connect to fully connected layers for classification or regression.

Inception-ResNet-v2 combines the advantages of both Inception modules and residual connections, offering robust feature extraction capabilities crucial for IDS applications. By leveraging its ability to capture multi-scale features and handle complex data dependencies, Inception-ResNet-v2 can analyze diverse network traffic data types effectively. It can be trained on labeled datasets of network behaviors to automatically learn and distinguish between normal traffic and various types of attacks, thereby enhancing the accuracy and reliability of intrusion detection systems in detecting and mitigating sophisticated threats.

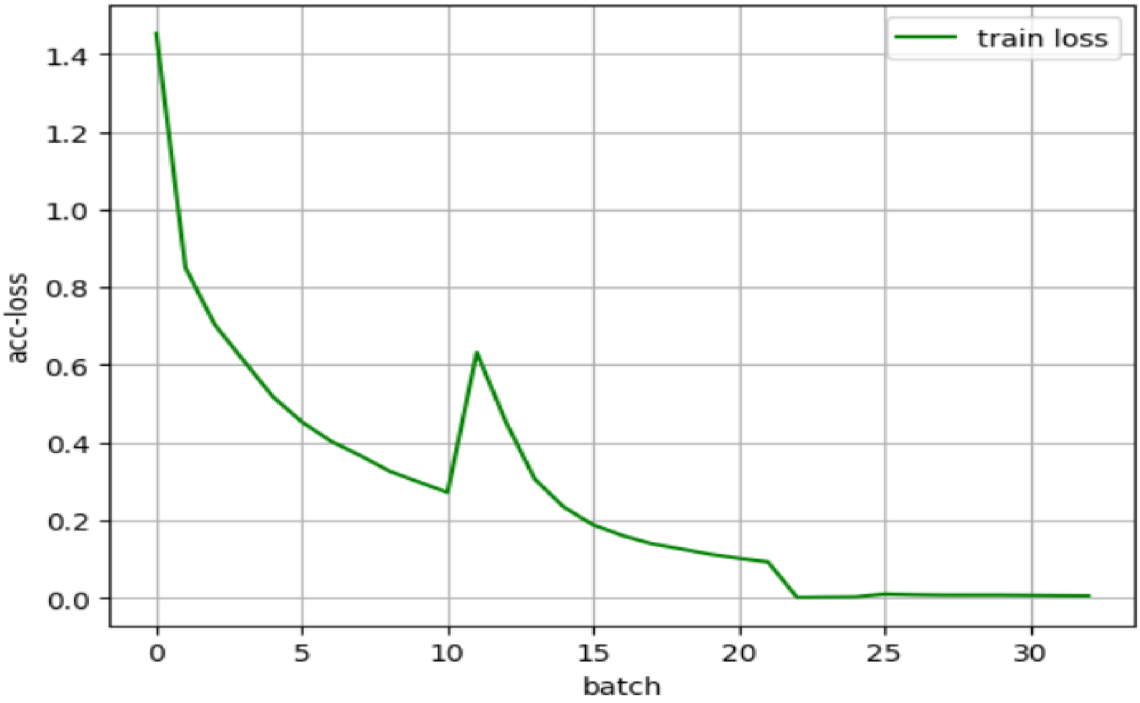


Fig. 4.8 : Training Accuracy-Loss Over Batches for Inception-ResNet Model

Acc-Loss vs Batch graph

Training Accuracy : 99.42%

Prediction Accuracy : 85.91%

JUSTIFCATION :

In comparison to other models used in Intrusion Detection Systems (IDS), Inception stands out due to its unique architecture that incorporates parallel convolutional operations and multi-scale feature extraction. While achieving a training accuracy of 98.86% and a prediction accuracy of 86.00%, Inception surpasses simpler architectures like VGG16 and VGG19, which typically exhibit strong performance but may not capture as nuanced patterns in network traffic data. Moreover, Inception-ResNet-v2, another advanced architecture, may offer similar or slightly improved performance due to its hybrid nature combining Inception and ResNet features, yet it often requires more computational resources. ResNet, known for its depth and resilience to vanishing gradients, excels in capturing complex dependencies but may not match Inception's ability to handle multi-scale features in IDS applications. Overall, Inception's balance between computational efficiency and accuracy makes it a compelling choice for IDS, particularly in detecting diverse and evolving network threats with high fidelity.

.

4.4. ATTACKS:

There are five types of Attacks we are detecting:

- **DoS Attack:**

  A Denial of Service (DoS) attack is a type of cyberattack where an attacker attempts to make a computer or network resource unavailable by overwhelming it with traffic or requests.

- **Fuzzy Attack:**

  A fuzzy attack is a type of cyberattack that exploits the vulnerabilities of complex systems, particularly in-vehicle networks.

- **Gear-Spoofing:**

  A gear spoofing attack is a type of cyberattack that targets the Controller Area Network (CAN) bus in vehicles. Specifically, it involves injecting false messages related to gear information into the CAN bus, which can cause the vehicle's systems to malfunction or behave erratically.

- **RPM-Spoofing:**

  An RPM (Revolutions Per Minute) spoofing attack is a type of cyberattack that targets the Controller Area Network (CAN) bus in vehicles. In this attack, malicious messages related to RPM information are injected into the CAN bus at a high frequency, typically every 1 millisecond. This can cause the vehicle's systems to malfunction or behave erratically, potentially leading to safety risks.

- **Normal(Attack-Free):**

  Attack-free state would refer to a situation where a system or network is not under attack or compromised by malicious activity.

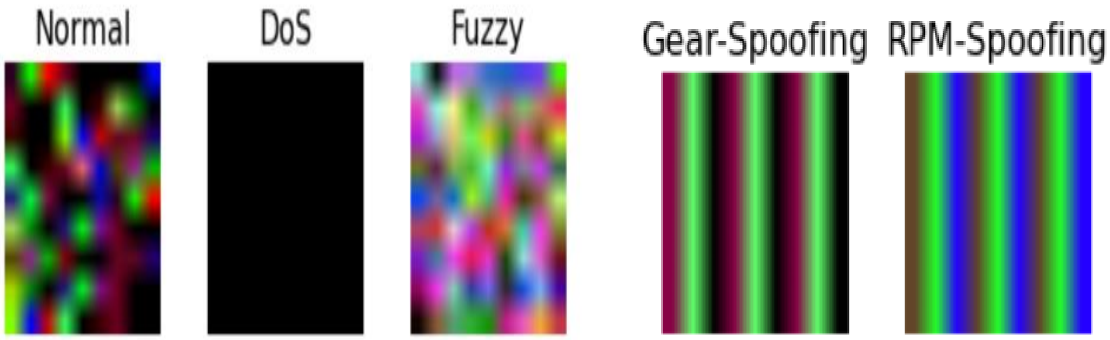


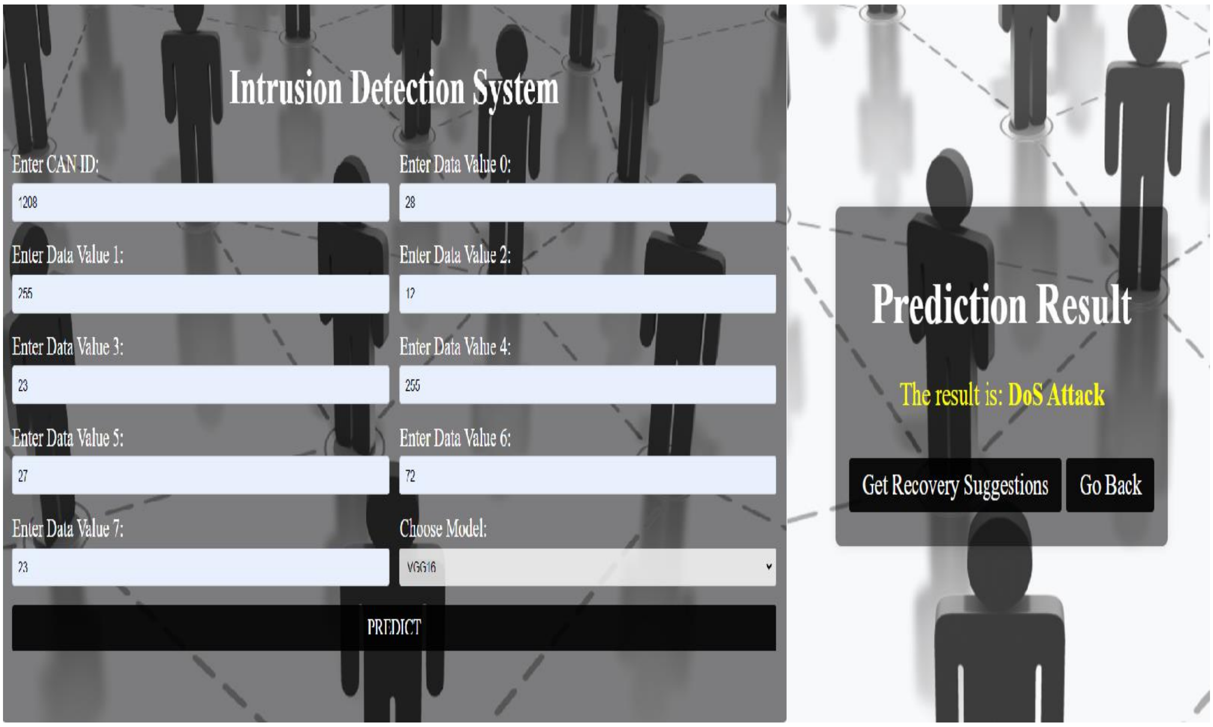
Fig.4.9 : Overview of Attack Patterns Detected by IDS

CHAPTER – 5

RESULTS AND DISCUSSION

5.1.RESULTS

5.1.1. IDS RESULTS



Intrusion Detection System

Enter CAN ID: 1208

Enter Data Value 0: 28

Enter Data Value 1: 255

Enter Data Value 2: 12

Enter Data Value 3: 23

Enter Data Value 4: 255

Enter Data Value 5: 27

Enter Data Value 6: 72

Enter Data Value 7: 23

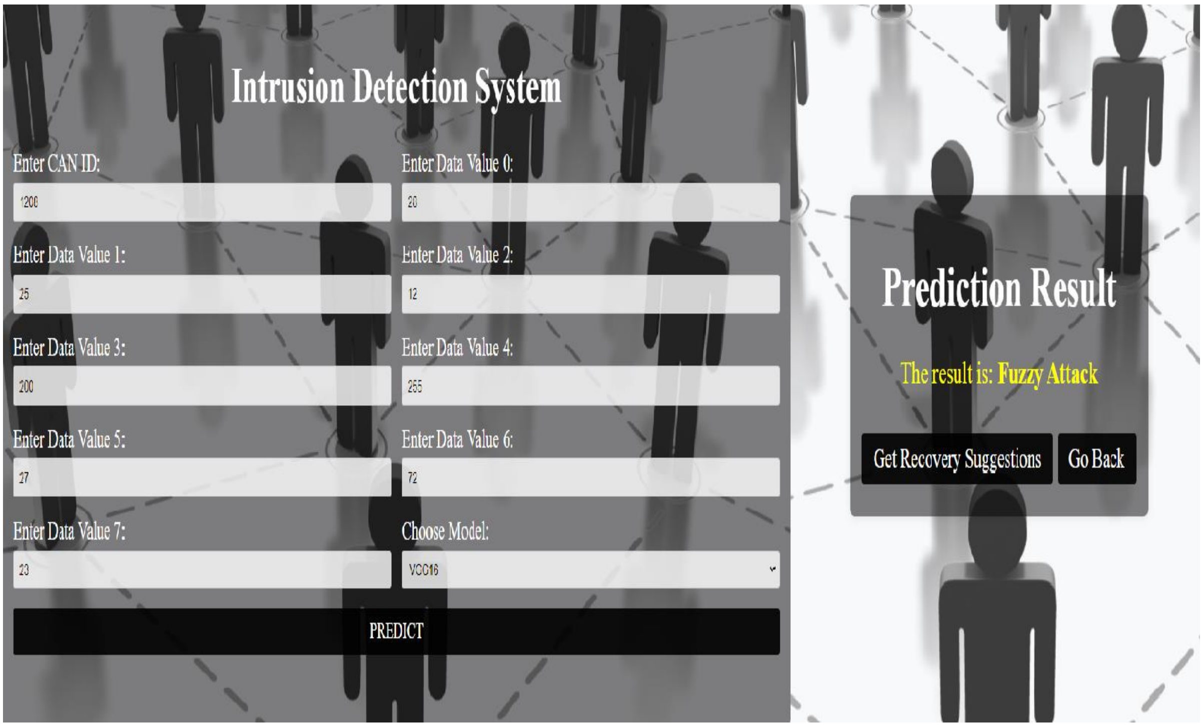
Choose Model: VGG16

Prediction Result

The result is: DoS Attack

Fig.5.1 : Predicting DoS Attacks with Intrusion Detection System

The system has analyzed the provided CAN message data and determined it to be indicative of a Denial of Service attack. This type of attack attempts to make a machine or network resource unavailable to its intended users, usually by temporarily or indefinitely disrupting services of a host connected to the Internet.



Intrusion Detection System

Enter CAN ID: 1200

Enter Data Value 0: 20

Enter Data Value 1: 25

Enter Data Value 2: 12

Enter Data Value 3: 200

Enter Data Value 4: 255

Enter Data Value 5: 27

Enter Data Value 6: 72

Enter Data Value 7: 23

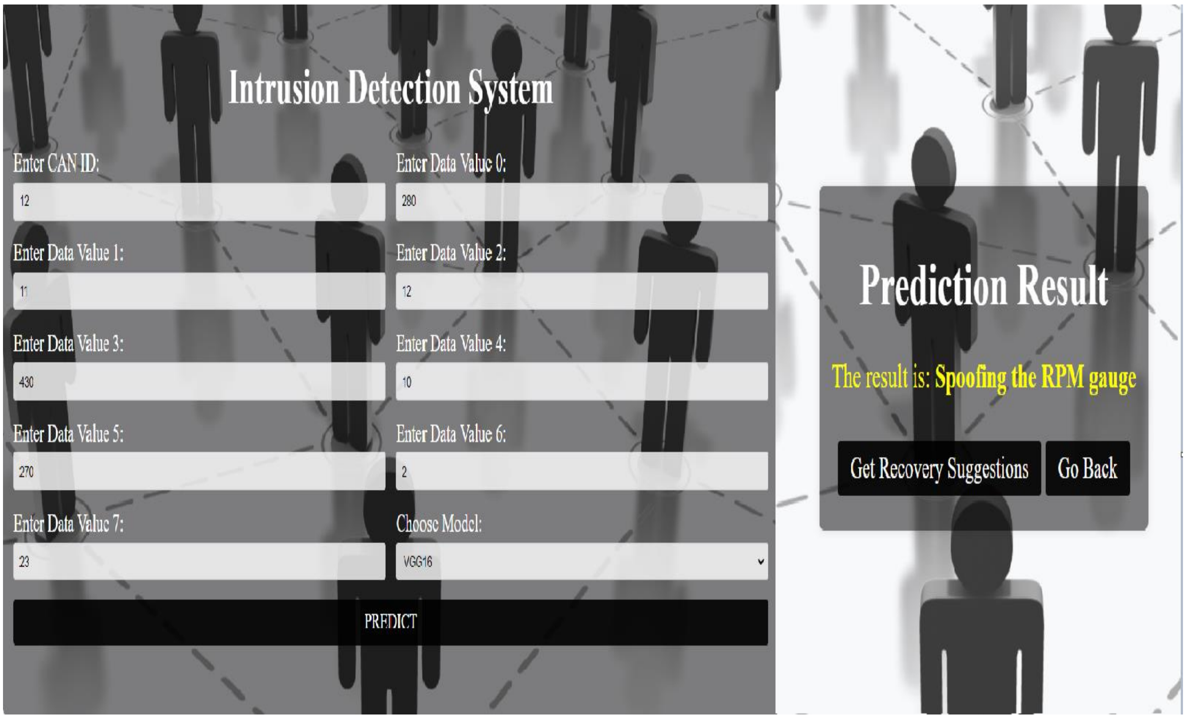
Choose Model: VGG16

Prediction Result

The result is: **Fuzzy Attack**

Fig.5.2 : Predicting Fuzzy Attacks with Intrusion Detection System

The system has determined the input CAN message data to be indicative of a Fuzzy attack. This type of attack typically involves sending random or unusual data to the system to find vulnerabilities. In CAN networks, it might mean sending unusual or malformed messages to discover how the system reacts.



Intrusion Detection System

Enter CAN ID: 12

Enter Data Value 0: 280

Enter Data Value 1: 11

Enter Data Value 2: 12

Enter Data Value 3: 430

Enter Data Value 4: 10

Enter Data Value 5: 270

Enter Data Value 6: 2

Enter Data Value 7: 23

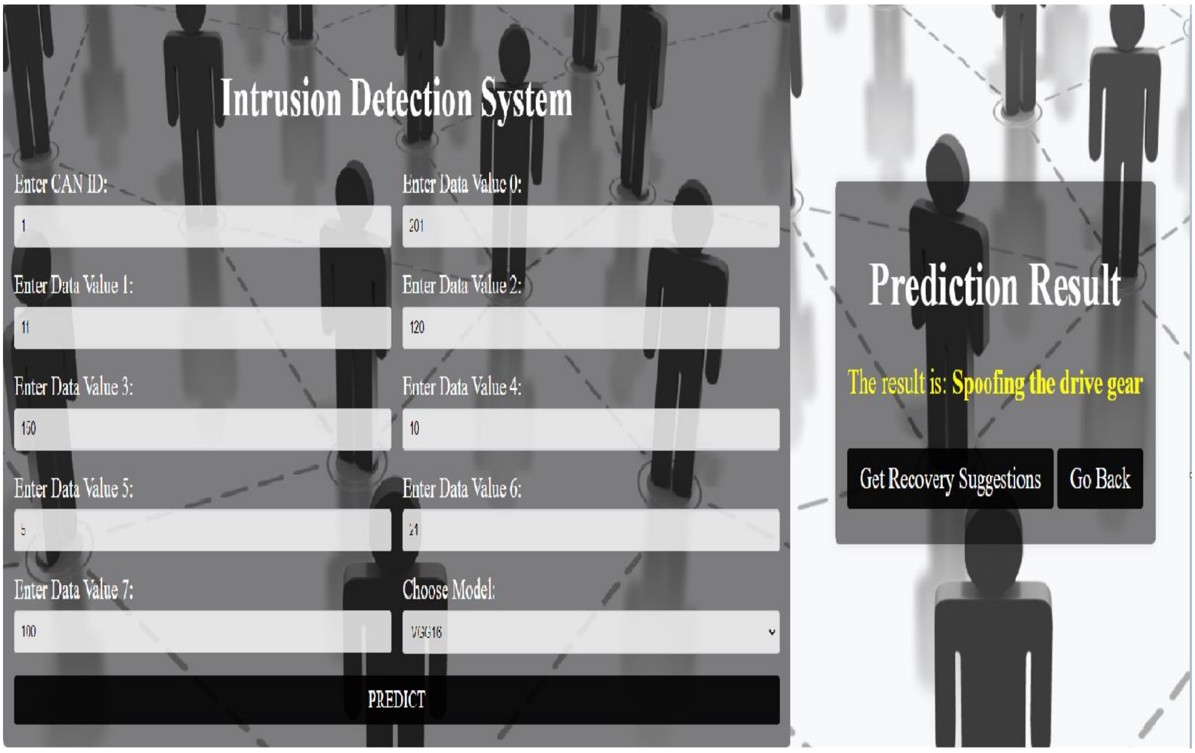
Choose Model: VGG16

Prediction Result

The result is: Spoofing the RPM gauge

Fig.5.3 : Predicting Spoofing the RPM gauge Attacks with Intrusion Detection System

The system has identified the input CAN message data as indicative of an attack aimed at spoofing the RPM (Revolutions Per Minute) gauge. This type of attack involves sending false data to the vehicle's system to manipulate the RPM gauge readings. Such attacks can mislead the driver about the engine speed and performance, potentially leading to unsafe driving conditions or damage to the vehicle.



Intrusion Detection System

Enter CAN ID: 1

Enter Data Value 0: 201

Enter Data Value 1: 11

Enter Data Value 2: 120

Enter Data Value 3: 150

Enter Data Value 4: 10

Enter Data Value 5: 5

Enter Data Value 6: 21

Enter Data Value 7: 100

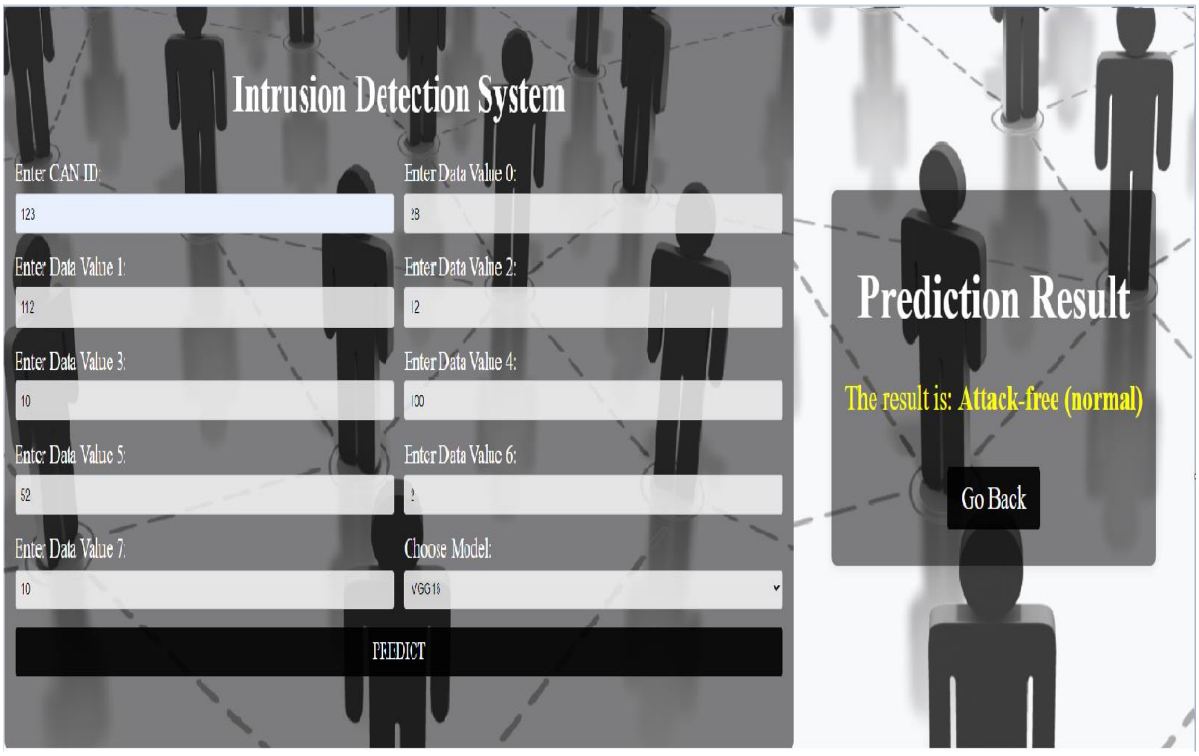
Choose Model: VGG16

Prediction Result

The result is: Spoofing the drive gear

Fig.5.4: Predicting Spoofing the drive gear Attacks with Intrusion Detection System

The result "Spoofing the drive gear" indicates that the Intrusion Detection System (IDS) has identified an anomaly characterized by the manipulation of drive gear information through the transmission of falsified data. In this context, the IDS has detected suspicious activities that suggest an attempt to deceive or mislead the system by altering the information related to the drive gear. Such spoofing attacks can undermine the integrity and reliability of critical system components, potentially leading to operational disruptions, safety risks, or unauthorized access.



Intrusion Detection System

Enter CAN ID: 123

Enter Data Value 0: 28

Enter Data Value 1: 112

Enter Data Value 2: 12

Enter Data Value 3: 10

Enter Data Value 4: 100

Enter Data Value 5: 52

Enter Data Value 6: 2

Enter Data Value 7: 10

Choose Model: VGG16

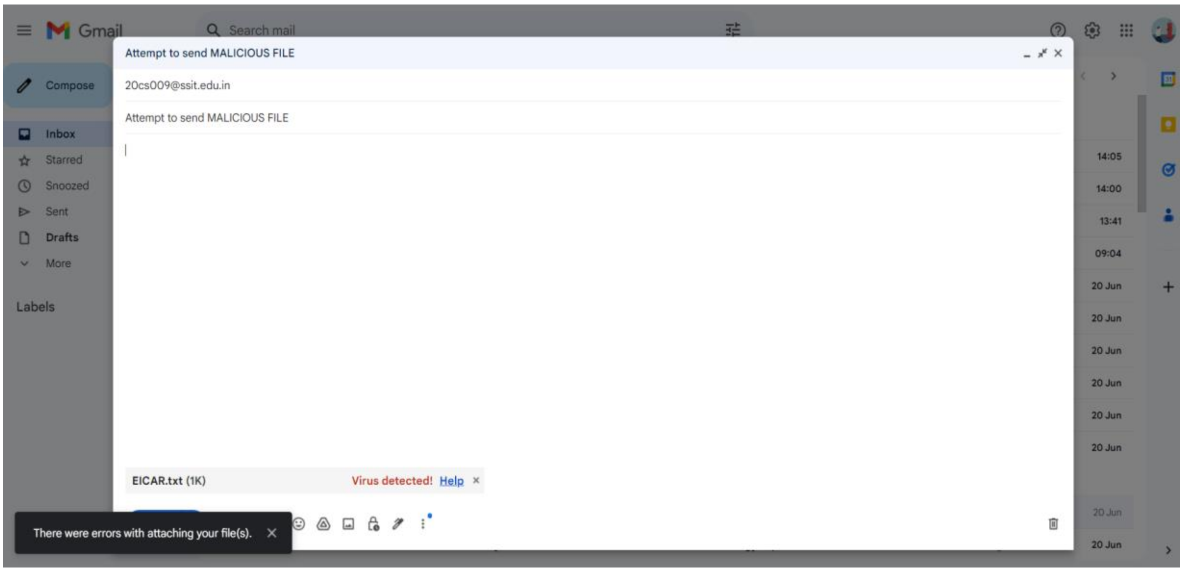
Prediction Result

The result is: Attack-free (normal)

Fig. 5.5: Predicting Attack-Free Status with Intrusion Detection System

The result "Attack-free (normal)" signifies that the Intrusion Detection System (IDS) has thoroughly examined the input data and concluded that there are no indications of any unauthorized or malicious activities. In this context, the IDS has meticulously analyzed the network traffic or system behavior against established patterns of normal operation. It has effectively recognized and validated that all activities observed align with expected norms and security policies, thereby confirming that the network or system is functioning without any intrusion attempts or security breaches.

5.2.1. IPS RESULTS



Attempt to send MALICIOUS FILE

20cs009@ssit.edu.in

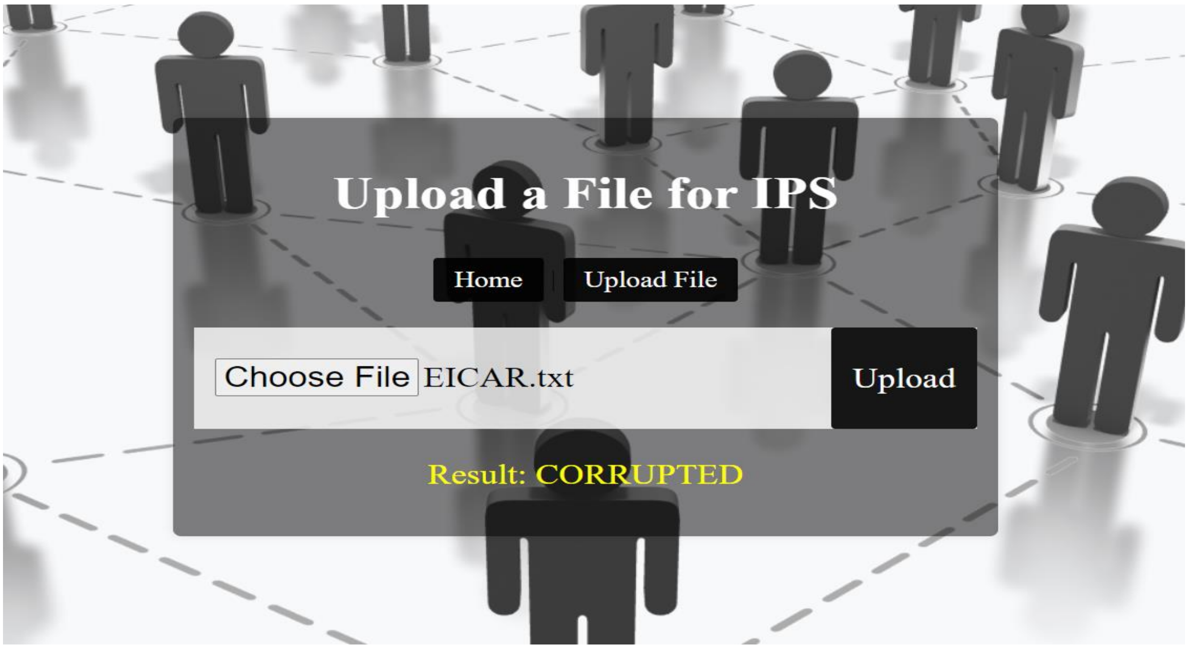
Attempt to send MALICIOUS FILE

EICAR.txt (1K) Virus detected! Help

There were errors with attaching your file(s).

Fig.5.7: Manual inspection of a suspicious file by forwarding it via email

This image demonstrates the process of manually inspecting a suspicious file by sending it through email for further examination.



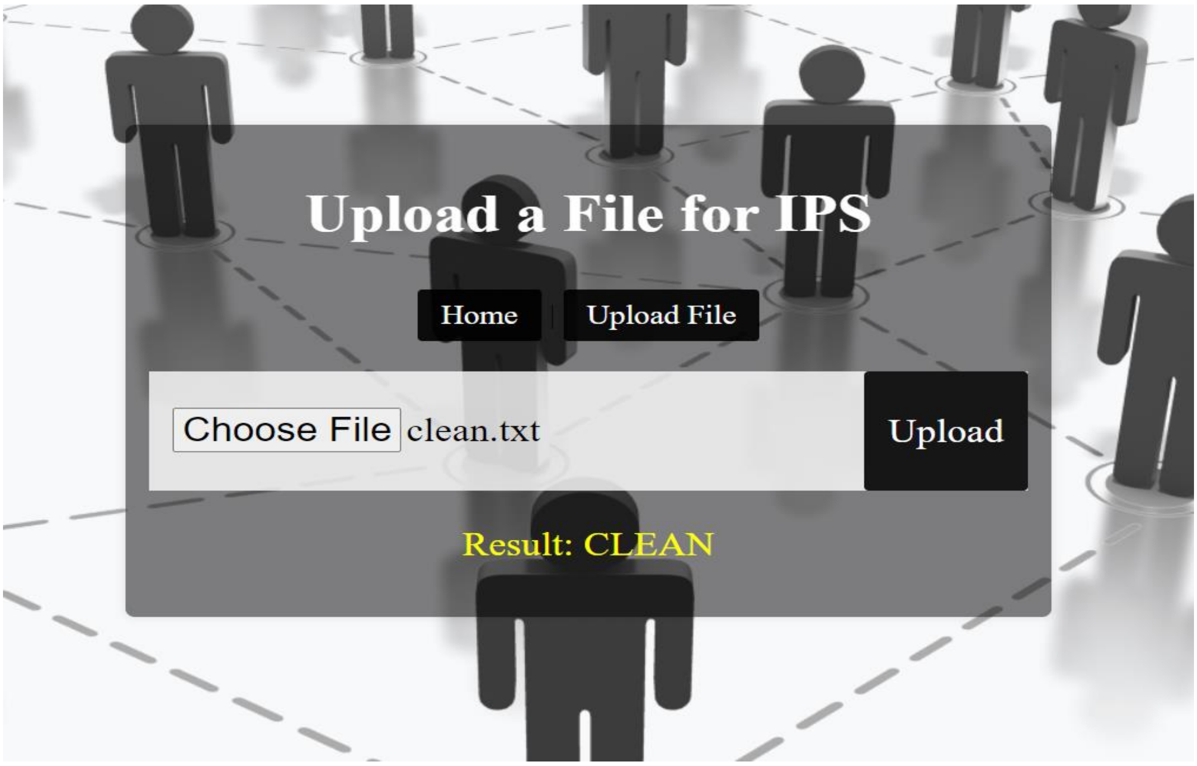
Upload a File for IPS

EICAR.txt

Result: CORRUPTED

Fig.5.8: Projecting corrupted file predictions through Intrusion Prevention Systems (IPS)

It illustrates an IPS's proactive role in identifying and forecasting corrupted files within a system or network. It likely displays graphical analyses and predictions of potential malware or malicious files. These visualizations are vital for administrators and security teams to grasp the threat landscape, take preemptive measures to block or neutralize threats, and uphold system integrity. By projecting and predicting corrupted file occurrences, the IPS plays a crucial role in preventing cybersecurity incidents and ensuring ongoing protection against evolving threats.



Upload a File for IPS

Choose File clean.txt

Result: CLEAN

Fig.5.9: Projecting clean file predictions through Intrusion Prevention Systems (IPS)

It shows how an IPS evaluates and predicts the presence of safe, non-malicious files within a network. It visually represents the IPS's capability to analyze files, assess their legitimacy based on predefined criteria, and forecast which files are deemed secure. This visualization helps administrators monitor system health, ensuring that only authorized files operate without posing cybersecurity risks. By projecting clean file predictions, the IPS plays a crucial role in preemptively safeguarding the network from potential threats, maintaining system integrity, and enabling proactive security measures.

5.2. SUMMARY

- Machine learning algorithms enhance the detection of novel and sophisticated cyber threats, using scikit-learn tools to train and deploy models for both IDS and IPS, ensuring high accuracy in identifying intrusions and malicious files.

- Flask provides an interactive web interface for real-time monitoring of network traffic and visualization of detected intrusions, offering timely alerts and actionable insights for swift response to security breaches.

- Feature selection and dimensionality reduction techniques improve the efficiency and accuracy of the IDS and IPS by streamlining data analysis, reducing complexity, and minimizing false positives.

- Text files are used to simulate malware behavior by embedding malicious patterns, aiding in the training and testing of the IPS to enhance detection and mitigation capabilities.

- The system dynamically updates its knowledge base to adapt to new and evolving threats, ensuring continuous learning and maintaining effectiveness against the latest cyber threats.

CHAPTER – 6

ADVANTAGES & APPLICATIONS

6.1. ADVANTAGES

- **Enhanced Detection Capabilities:**

By integrating machine learning algorithms, the system can detect both known and unknown threats, including zero-day attacks, which traditional rule-based systems might miss.

- **Real-time Monitoring and Response:**

The system's ability to provide real-time monitoring and visualization ensures that network administrators can quickly identify and respond to potential security incidents, minimizing damage.

- **Reduced False Positives:**

Feature selection and dimensionality reduction techniques help streamline data analysis, improving the accuracy of threat detection and significantly reducing the number of false positives.

- **Adaptability and Continuous Learning:**

The system can dynamically update its knowledge base, adapting to new and evolving threats to maintain effective defense over time.

- **Comprehensive Coverage:**

The combination of IDS and IPS provides a multi-layered security approach, covering both network traffic analysis and file-based threat detection for a more holistic defense strategy.

- **Scalability:**

The system is designed to be scalable, making it suitable for deployment in diverse network environments, from small businesses to large enterprises.

6.2. APPLICATIONS

- **Enterprise Network Security:**

Large organizations can deploy the IDS and IPS to protect their networks from a wide range of cyber threats, ensuring the integrity and confidentiality of sensitive data.

- **Cloud Security:**

Cloud service providers can integrate this system to monitor and secure cloud environments, protecting against intrusions and malicious activities targeting cloud-based resources.

- **Financial Institutions:**

Banks and financial institutions can use the system to safeguard against cyber attacks that aim to steal sensitive financial information or disrupt operations.

- **Healthcare Sector:**

Hospitals and healthcare providers can deploy the system to protect patient data and ensure compliance with regulatory standards such as HIPAA, preventing data breaches and unauthorized access.

- **Government and Defense:**

Government agencies and defense organizations can use the IDS and IPS to protect critical infrastructure and sensitive information from cyber espionage and attacks.

- **Educational Institutions:**

Universities and research institutions can implement the system to protect academic data, research projects, and personal information of students and staff from cyber threats.

- **Small and Medium-sized Enterprises (SMEs):**

SMEs can benefit from the scalable nature of the system, enabling them to protect their networks with advanced security measures typically available to larger organizations. This allows SMEs to maintain a high level of security without the need for extensive in-house expertise or resources. The system's user-friendly interface and automated features make it accessible for smaller teams, ensuring robust protection against cyber threats with minimal management overhead. Additionally, by preventing potential breaches and associated costs, SMEs can safeguard their reputation and customer trust, which is crucial for business growth and sustainability.

CHAPTER – 7

CONCLUSION

7.1. CONCLUSION

In conclusion, the proposed intelligent Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) represent a significant advancement in the field of cybersecurity. By integrating traditional rule-based methods with advanced machine learning techniques, the system provides a robust and adaptive defense against a wide array of cyber threats. The use of machine learning algorithms from scikit-learn enhances the system's ability to detect novel and sophisticated attacks that traditional methods might miss. Real-time monitoring and visualization through Streamlit ensure that administrators have immediate insights into network activity and can respond swiftly to potential security breaches.

The implementation of feature selection and dimensionality reduction techniques has proven effective in improving the efficiency and accuracy of the IDS and IPS. These techniques help streamline data analysis, reducing the complexity and computational load while minimizing false positives. Additionally, the use of text files for malware simulation has facilitated the training and testing of the IPS, enabling it to recognize and mitigate malicious behaviors effectively. Continuous learning and adaptation are key strengths of the proposed system. By dynamically updating its knowledge base, the IDS and IPS remain effective against evolving threats, ensuring long-term protection. Rigorous testing in controlled environments has validated the system's robustness and reliability, demonstrating its capability to safeguard networks against a wide range of cyber threats.

Overall, this project highlights the potential of combining machine learning with traditional security measures to create a comprehensive and adaptive security solution. The intelligent IDS and IPS not only enhance network resilience but also provide a scalable framework that can be adapted to diverse network environments. The successful implementation and testing of this system underscore its value as a critical component of modern cybersecurity strategies.

7.2. FUTURE WORK

While the proposed IDS and IPS have demonstrated significant capabilities, several areas for future enhancement can further refine the system. Expanding the dataset to include a broader range of network traffic patterns and attack types will improve the system's generalization and detection accuracy. Incorporating deep learning techniques, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), can further enhance the system's ability to detect complex patterns and anomalies in network traffic. Integrating real-time threat intelligence feeds will keep the system updated on emerging threats, ensuring proactive defense capabilities.

Additionally, developing more sophisticated anomaly detection algorithms will reduce false positives and improve overall accuracy. Implementing automated incident response actions, such as quarantining affected devices or blocking malicious IP addresses, can significantly reduce response times and mitigate the impact of detected threats. Finally, extensive field testing in real-world network environments will provide valuable insights into the system's practical effectiveness and highlight areas for further optimization. These efforts will ensure that the intelligent IDS and IPS remain at the forefront of cybersecurity defense.

REFERENCES

[1] Almutairi, A.H., Abdelmajeed, N.T., Innovative signature based intrusion detection system: Parallel processing and minimized database, in: 2017 International Conference on the Frontiers and Advances in Data Science (FADS), IEEE. pp. 114–119. [2017].

[2] S. Walker-Roberts, M. Hammoudeh, O.Aldabbas, M. Aydin, and A. Dehghantanha , Threats on the horizon:Understanding cyber-physical systems. Heliyon 5, e01802. [2018].

[3] J.-P. A. Yaacoub, O. Salman, H. N. Noura, N. Kaaniche, A. Chehab, and M. Malli . Cyber-physical systems security:Limitations, issues and future trends.[2020].

[4] Ammar, A., et al., A decision tree classifier for intrusion detection priority tagging. Journal of Computer and Communications 3, 52. [2015].

[5] J. Preden, “Generating situation awareness in cyberphysical systems: Creation and exchange of situational information,IEEE [2020].

[6] Q. V. Le and T. Mikolov, Distributed representations of sentences and documents,IEEE, pp. 152-156 [2007].

[7] Gascon, H., Orfila, A., Blasco, Analysis of update delays in signature-based network intrusion detection systems. Computers Security 30, 613–624 [2011].

[8] Friedman, J.H.,Greedy function approximation: a gradient boosting machine. Annals of statistics , 1189–1232 [2001].