Project Title : Network Intrusion Detection System Using Deep learning

Synopsis submitted to

Shri Ramdeobaba College of Engineering & Management, Nagpur

in partial fulfillment of requirement for the award of

degree of

Bachelor of Engineering

In

COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

By

Ms. Khushi Kharate

Ms. Panchami Vishwakarma

Ms. Rama Labhe

Mr. Aryaman Pandey

Guide

Dr. Yogesh Thakare

Computer Science and Engineering

Shri Ramdeobaba College of Engineering & Management, Nagpur

440010

Network Intrusion Detection System

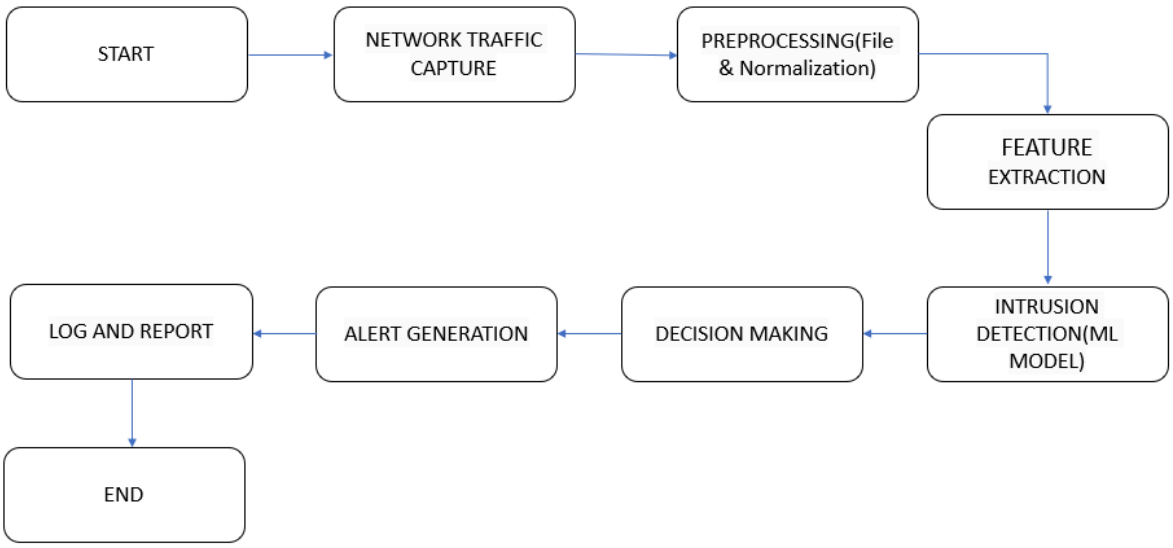
Project Definition:

The project addresses the critical challenge of protecting network infrastructure from malicious activities by providing real-time detection and classification of potential intrusions. As network environments grow increasingly complex, traditional security measures often fall short in identifying sophisticated threats and anomalies, leaving systems vulnerable to attacks. This project aims to develop a comprehensive Network Intrusion Detection System (NIDS) using advanced machine learning models to accurately detect and classify network intrusions and anomalies in real-time.
By seamlessly integrating this system with existing network security infrastructure, it enhances overall network protection by enabling automated threat response and proactive security measures. This unified approach mitigates risks associated with undetected intrusions, contributing to a more secure and resilient network environment.

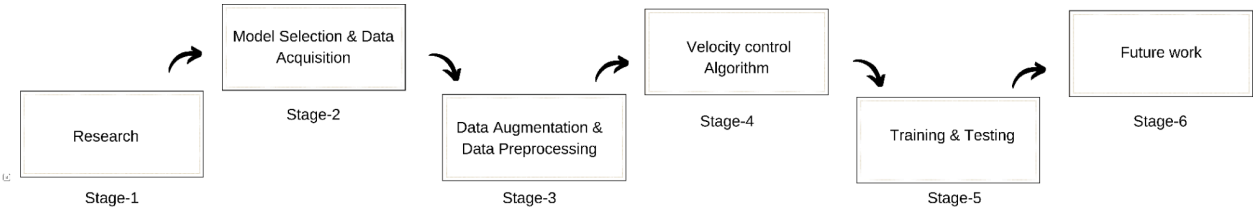
Project Objective:

- **Real-Time Monitoring:** Implement continuous monitoring of network traffic to detect and alert on suspicious activities as they occur . Utilize advanced algorithms to ensure timely detection and immediate alerting for potential threats.
- **High Detection Accuracy:** Achieve high accuracy in distinguishing between legitimate and malicious activities, reducing false positives and false negatives . Employ machine learning techniques and threat intelligence to enhance detection capabilities.
- **Scalability:** Ensure the system can scale to handle varying network sizes and large volumes of traffic without performance degradation . Design the architecture to support distributed environments and cloud-based infrastructures.
- **Comprehensive Threat Detection:** Develop mechanisms to identify a wide range of threats, including known vulnerabilities and emerging, unknown threats . Incorporate signature-based, anomaly-based, and behavior-based detection methods.
- **User-Friendly Interface:** Provide an intuitive interface for administrators to monitor network activity, analyze alerts, and manage the system efficiently . Include customizable dashboards and detailed reporting features for comprehensive oversight.
- **Regular Updates:** Implement a framework for regular updates and improvements to keep the NIDS current with evolving threat landscapes . Establish a process for timely integration of new threat signatures and detection algorithms.

Flowchart:



Proposed Plan of Work:



Proposed Plan of Work

1. Data Acquisition: Collect and preprocess the NSL-KDD dataset.Split data into training, validation, and test sets.

2. Model Development:

- **Deep Learning:**Train Autoencoders, LSTMs, and MLPs for anomaly detection and classification.
- **Traditional ML:** Implement KNN, LDA, SVM, and QDA for enhanced accuracy.

3. Velocity Control Algorithm: Develop and integrate an algorithm for dynamic speed limit adjustment based on road curvature.Implement safety features for warning users of potential hazards.

4. Testing and Validation: Evaluate models with accuracy, precision, recall, F1-score, and ROC-AUC .Perform cross-validation and system testing to ensure robustness and accuracy.

5. User Interface Development: Design and develop an intuitive UI for system interaction

and real-time data visualization.Ensure seamless integration with backend models and conduct usability testing.

Methodology:

Model Development: Use Autoencoders, LSTMs, and MLPs for anomaly detection and classification.Implement KNN, LDA, SVM, and QDA for enhanced classification accuracy.

Data Preparation: Utilize the NSL-KDD dataset.Preprocess data: handle missing values, normalize, and encode features.

Model Evaluation: Assess models using metrics like accuracy, precision, recall, F1-score, and ROC-AUC.Perform cross-validation to ensure model robustness.

Development Environment: Use Google Colaboratory and Jupyter Notebook for model development and testing.

Sr No	Steps	Time
1	Gathering information	1 week (Done)
2	Data Acquisition	3 days (In progress)
3	Model Selection	1 week (In progress)
4	Data preprocessing	3 days
5	Training and Testing / evaluation	1 week
6	Developing Model	1 week
7	Deploying on website	4 days

Technology Used:

1. Programming Languages and Libraries: Python,Keras,Scikit-learn (Sklearn),Pandas,NumPy,Matplotlib,Pickle

2. Machine Learning Models:

- **Autoencoder:** Unsupervised learning, anomaly detection.
- **LSTM:** Sequence modeling, temporal dependencies.
- **MLP:** Complex classification.
- **KNN:** Instance-based classification**.**
- **LDA:** Dimensionality reduction, classification.

- **SVM:** Optimal class separation.
- **QDA:** Flexible classification, varying covariance matrices.

3. Data Handling and Visualization:

- **CSV/Text Files:** Storing and managing datasets.
- **Numpy Arrays:** Label storage.
- **Plots:** Visualizing model performance (accuracy, loss).

4. Development Environments:

- **Google Colaboratory:** Cloud-based notebook environment with GPU support.
- **Jupyter Notebook:** Local environment for running and developing notebooks.

5. Data Source:

- **NSL-KDD Dataset:** Updated dataset used for training and evaluation.

Functional Specification (Deliverables):

1. Real-Time Intrusion Detection: The system will accurately detect potential network intrusions in real-time. Detection shall be provided continuously to ensure timely identification and response to potential threats.

2. Intrusion Classification: Utilizing machine learning models such as Random Forest, SVM, or Deep Learning models, the system will classify different types of network intrusions with high accuracy and reliability. Classification shall be performed continuously to ensure the system remains updated with evolving threats.

3. Anomaly Detection: Autoencoders or other unsupervised learning techniques can be employed to detect anomalies in network traffic. Anomaly detection shall be dynamically updated to reflect changes in network behavior.

4. Alert Mechanism: The system will dynamically generate alerts based on detected intrusions and anomalies. Alerts shall be displayed to the user through a user interface or notification system in real-time.

5. Integration with User Interface: The system shall integrate seamlessly with existing network monitoring interfaces or dashboards. Intrusion information shall be displayed clearly and intuitively to users, ensuring ease of understanding.

6. Threat Response Mechanism: In case of detected intrusions, the system shall issue warnings to alert users or network administrators. Warnings shall be displayed prominently and accompanied by audible alerts or notifications to ensure timely response.

7. Compliance and Security: The system shall adhere to industry standards and best practices for network security and data privacy. It should ensure the confidentiality, integrity, and availability of network data while complying with relevant regulations.

Project Scope:

- **System Development:T**he project will involve the design and development of a comprehensive system capable of providing real-time information to network administrators regarding potential intrusions and anomalies in network traffic.
- **Machine Learning Integration:**Integration of advanced machine learning models, including Random Forest, SVM, Autoencoders, and CNNs, to detect and classify network intrusions, identify anomalies, and dynamically update security measures in real-time.
- **User Interface Development:**Design and development of user interfaces, including dashboards and notification systems, to present intrusion detection information and alerts to network administrators in a clear and intuitive manner.
- **Network Security Integration:**Seamless integration of the developed system with existing network security infrastructure to enable automated threat response and proactive security measures. The system will enhance overall network protection by working in conjunction with firewalls, IDS/IPS, and other security tools.

Conclusion:-

This project leverages a blend of deep learning and traditional machine learning techniques to create a robust Network Intrusion Detection System. It employs Autoencoders, Long Short-Term Memory (LSTM) networks, and Multi-Layer Perceptrons (MLPs) for effective anomaly detection and classification across both binary and multi-class tasks. To further enhance accuracy, the system integrates K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA), Support Vector Machines (SVM), and Quadratic Discriminant Analysis (QDA). Development is carried out using Google Colaboratory and Jupyter Notebook, while the NSL-KDD dataset provides comprehensive data for training and evaluation. The result is a sophisticated system capable of accurate and efficient network intrusion detection, aimed at bolstering network security and reliability.

Team member details:

Roll No.	Name of the Students	Name of the Guide
06	Khushi Kharate	Dr. Yogesh Thakre
14	Panchami Vishwakarma	
16	Rama Labhe	
22	Aryaman Pandey	