DETECTING CYBER THREATS THROUGH ANOMALY DETECTION IN NETWORK TRAFFIC DATA

Student Name: Harish.K

Register Number: 511523205017

Institution: P.T.Lee Chengalvaraya Naicker College Of Engineering and Technology

Department: Information Technology

Date of Submission: 2025-04-26

1. Problem Statement

Cybersecurity threats continue to rise in frequency and sophistication, putting organizations at risk. Traditional security systems often fail to detect novel or subtle attacks. This project aims to detect cyber threats using anomaly detection techniques applied to network traffic data, enabling the identification of suspicious behaviors that may indicate a breach.

2. Objectives of the Project

- Analyze network traffic data to understand normal vs. abnormal behavior.
- Build models to detect anomalies that may signal cyber threats.
- Visualize detection results and trends.
- Provide insights for timely threat response.

3. Scope of the Project

The project includes analyzing network traffic logs and applying unsupervised or semi-supervised learning techniques to detect anomalies. Constraints include the availability of labeled datasets and the focus on detection rather than prevention or response mechanisms.

4. Data Sources

Publicly available network traffic datasets from sources like Kaggle, CICIDS (Canadian Institute for Cybersecurity), or UNSW-NB15. These are static datasets with labeled or unlabeled traffic logs.

5. High-Level Methodology

- Data Collection: Obtain datasets from public sources.

- Data Cleaning: Handle missing or corrupted entries and standardize formats.

- Exploratory Data Analysis: Use time-series plots and distribution graphs to explore traffic behavior.

- Feature Engineering: Extract protocol types, packet sizes, duration, and connection patterns.

- Model Building: Use algorithms like Isolation Forest, One-Class SVM, or Autoencoders.

- Model Evaluation: Use precision, recall, ROC-AUC, and confusion matrix.

- Visualization & Interpretation: Graphical representation of normal vs. anomalous activities.

- Deployment: (Optional) Build a prototype dashboard using Streamlit or Gradio.

6. Tools and Technologies

- Programming Language: Python

- Notebook/IDE: Jupyter Notebook, Google Colab

- Libraries: pandas, numpy, matplotlib, seaborn, scikit-learn, keras, pyOD

- Optional Tools for Deployment: Streamlit, Gradio

7. Team Members and Roles

- Data Preparation and Cleaning:Adhithya.A

- Feature Engineering and EDA:Harisharan.p.s

- Anomaly Detection Modeling:Harish.k

- Results Interpretation and Reporting:Prasanth.p