**Problem Statement:**

The project aims to detect anomalies in network traffic (e.g., DDoS attacks) using machine learning. The focus of this phase is on **exploratory data analysis (EDA)** and building a **baseline model**.

**Dataset Used:**

**CICIDS 2017 - Friday Working Hours Afternoon (DDoS attack)**

* Contains labeled network traffic with both benign and malicious (DDoS) flows.
* Features include flow duration, packet sizes, inter-arrival times, etc.

**Data Cleaning Steps:**

* Replaced "Infinity" and similar string values with np.nan.
* Converted all appropriate features to numeric.
* Removed columns with all missing or constant values.
* Dropped a few features with extremely large values (e.g., Flow Bytes/s, Fwd IAT Total) which were distorting scaling and multicollinearity.

**Exploratory Data Analysis:**

* **Correlation Heatmap** revealed highly correlated IAT-based and flow size features.
* Used **boxplots** to visualize how traffic features differ between benign and attack traffic.
* Label distribution was relatively balanced: ~57% attack, ~43% benign.

**Feature Engineering:**

* Label column was converted to binary (BENIGN = 0, everything else = 1).
* Scaled features using StandardScaler.

**Baseline Model: Logistic Regression**

*LogisticRegression(max\_iter=1000)*

**Evaluation Results:**

|  |  |
| --- | --- |
| Metric | Score |
| Accuracy | 99.85% |
| Precision | 1.00 |
| Recall | 1.00 |
| F1-Score | 1.00 |
| ROC AUC Score | 0.9998 |

**Confusion Matrix**: Very few false positives/negatives (67 total misclassifications out of 45,000+ samples).

**ROC Curve**: Hugging top-left, indicating excellent discrimination ability.

**Next Steps (Module 24):**

* Experiment with more advanced models like **Random Forest**, **Isolation Forest**, or **Autoencoders**.
* Perform **hyperparameter tuning** and **cross-validation**.
* Possibly integrate live traffic data collection via Zeek/Wireshark.
* Build a dashboard or alerting system (AWS/Docker) for real-time anomaly detection.