**AI-based Threat Detection using CICIDS2017 Dataset and MITRE ATT&CK Mapping**

**Introduction**

This project focuses on detecting cyber threats using machine learning models trained on the CICIDS2017 dataset. It integrates MITRE ATT&CK mapping to associate detected threats with known adversarial techniques. The goal is to demonstrate how supervised machine learning can classify network intrusions and relate them to real-world tactics and techniques.

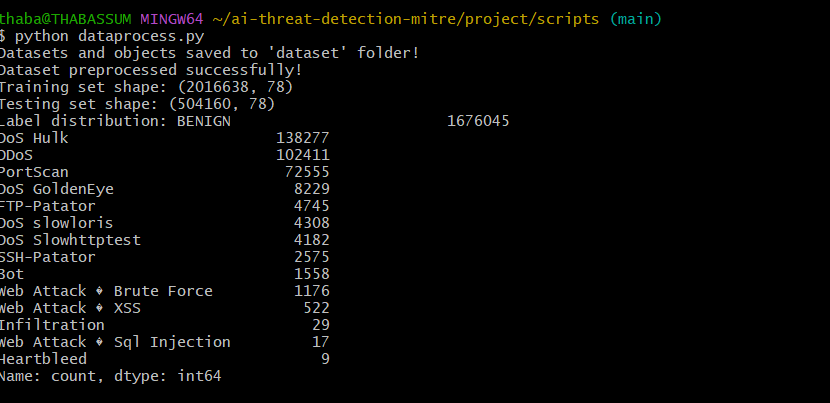
**Project Workflow Overview**

The project is divided into the following phases:

1. Data Preprocessing
2. Model Training (Random Forest)
3. Model Evaluation
4. Confusion Matrix Visualization
5. MITRE ATT&CK Mapping and Frequency Visualization

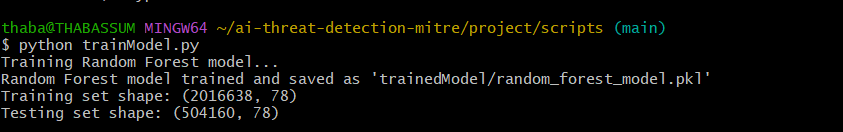
**1. Data Preprocessing**

* Loaded multiple daily CSV logs from CICIDS2017 dataset.
* Cleaned the dataset:
  + Removed leading/trailing spaces in column names
  + Replaced infinite values with NaN
  + Dropped all rows with NaN values
  + Removed duplicate entries



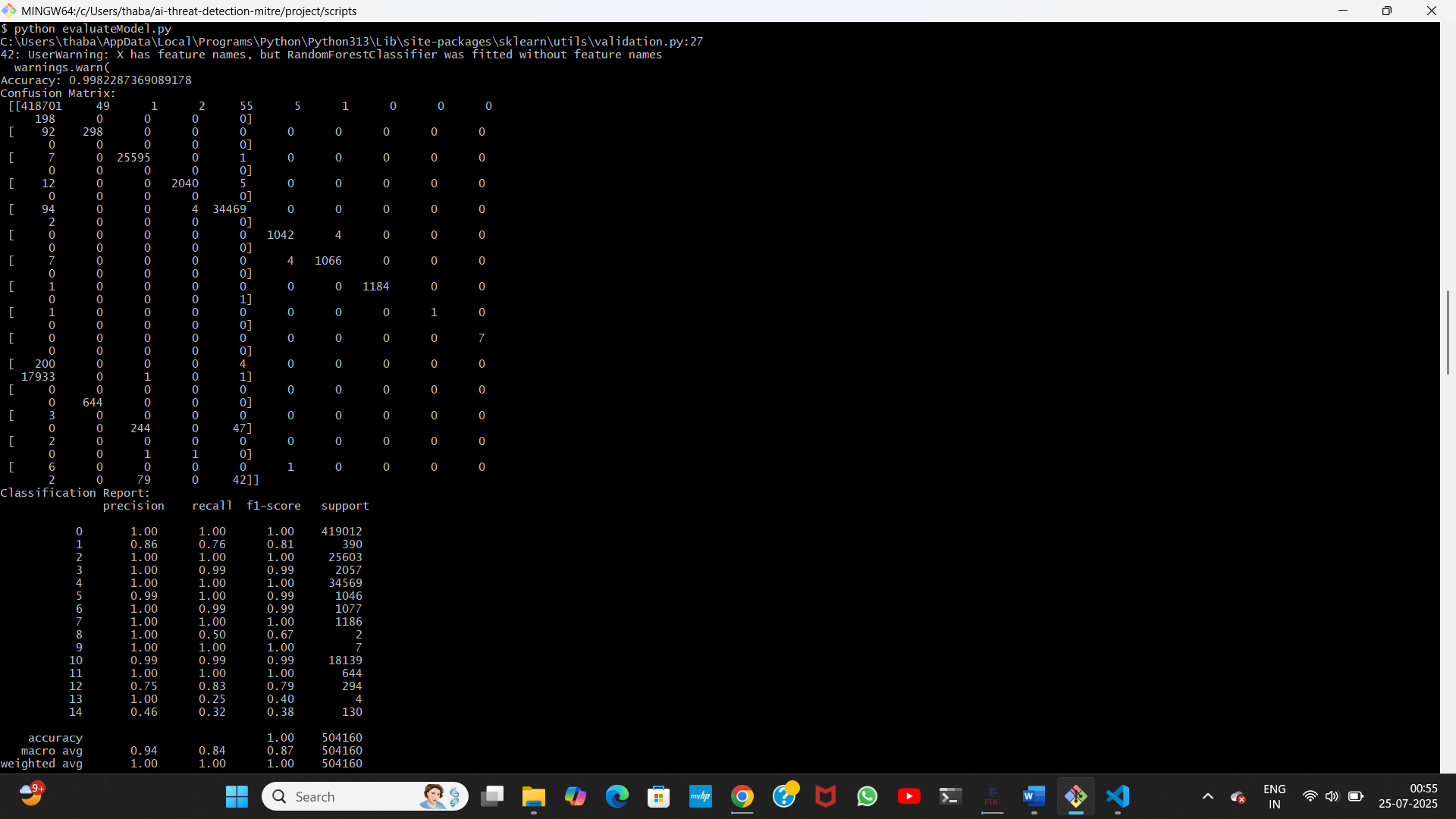
**2. Model Training**

* Utilized a RandomForestClassifier with n\_estimators=100, and random\_state=42
* Trained on preprocessed feature set using train\_test\_split
* Saved the trained model to a .pkl file



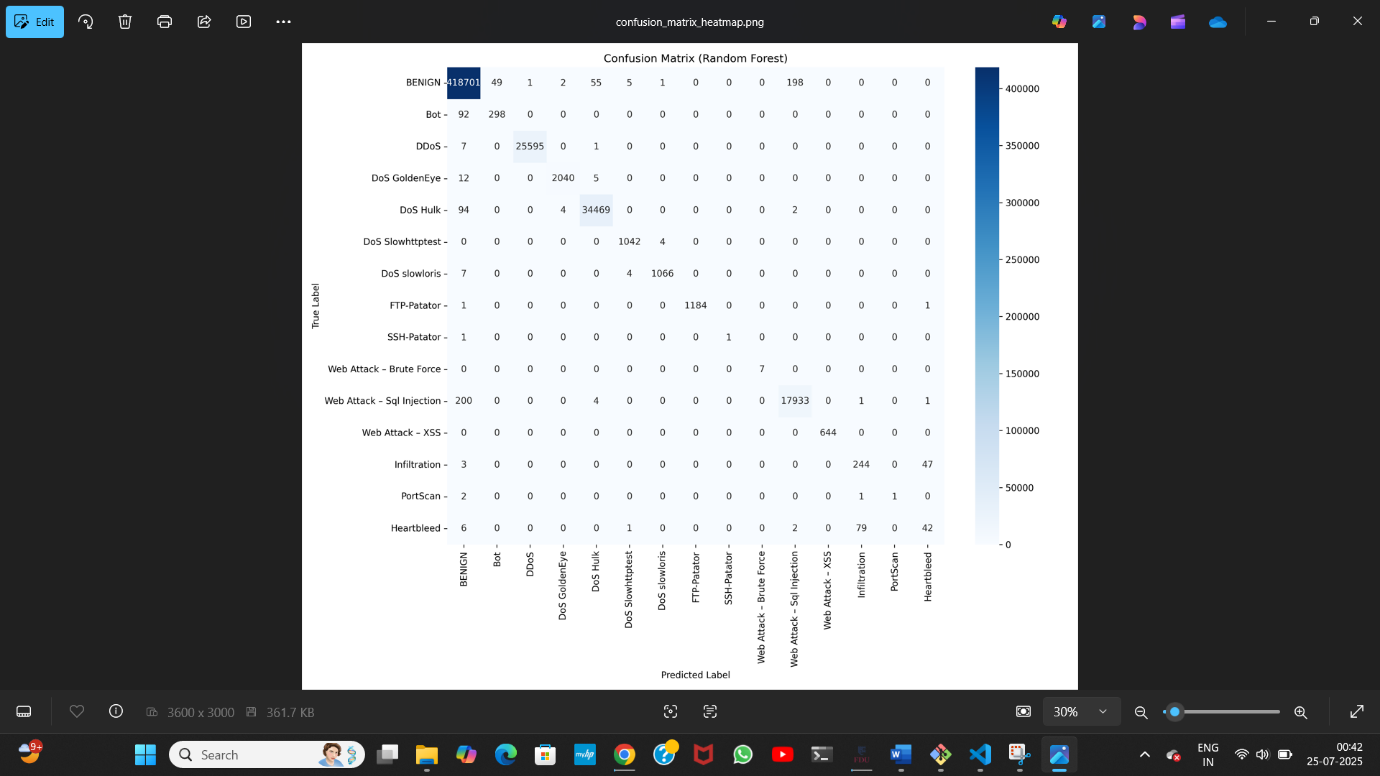
**3. Model Evaluation**

* Used accuracy score, confusion matrix, and classification report
* Saved results to evaluation\_results.csv



**4. Confusion Matrix**

* Confusion matrix plotted using seaborn heatmap
* Displays class-wise classification accuracy



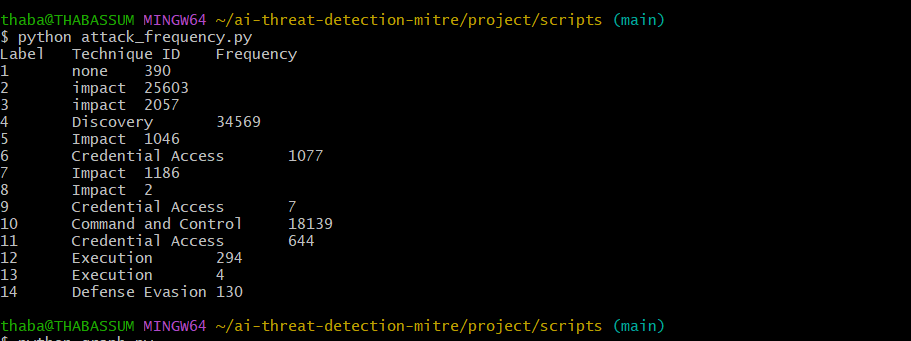
**5. Attack Frequency Mapping**

* Labels from dataset were mapped to MITRE Technique IDs via a custom mapping CSV (attack\_to\_mitre\_mapping.csv)

图形用户界面, 表格

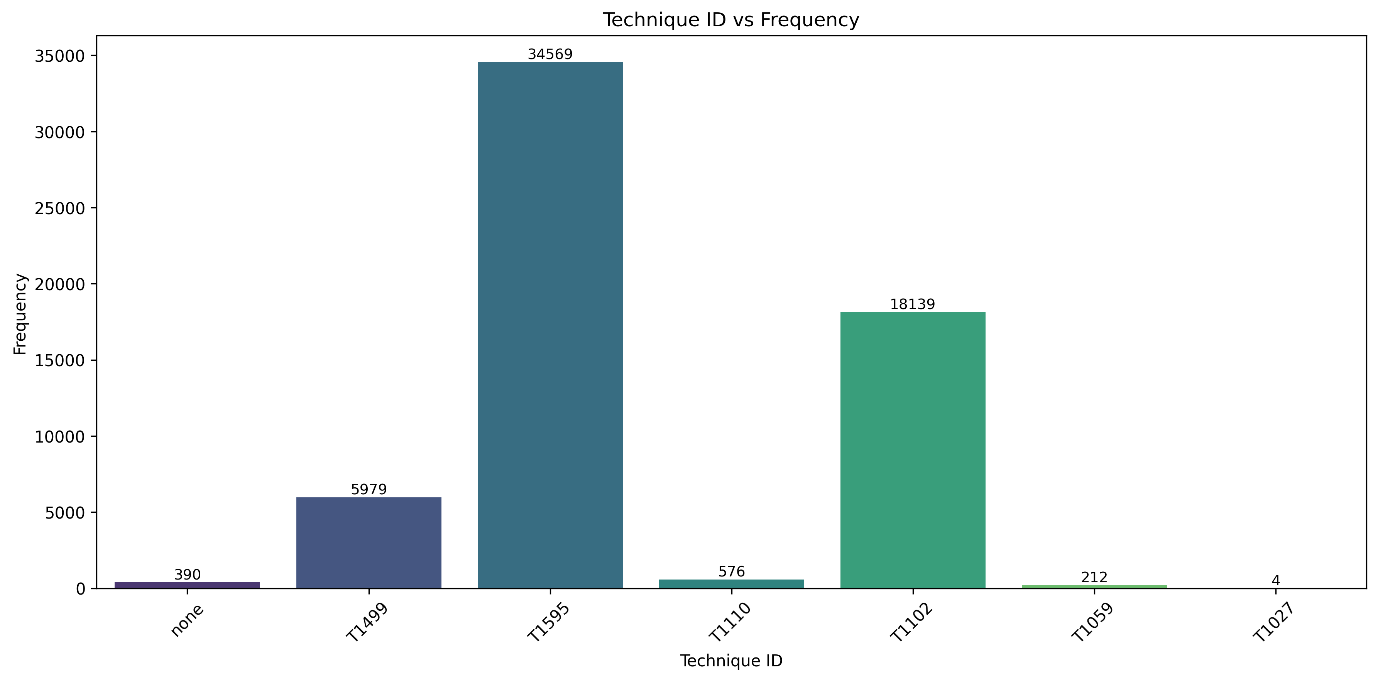
AI 生成的内容可能不正确。

* Frequencies of each attack technique were computed



**6. Graphical Representation of MITRE Techniques**

* Plotted a bar chart using seaborn
* X-axis: Technique ID
* Y-axis: Attack frequency



Attack heatmap:

图片包含 条形图

AI 生成的内容可能不正确。

**Key Findings and Conclusion**

* The model achieved over 99% accuracy, which is unusually high.

Such a result may indicate overfitting:

* The model performs extremely well on this dataset but may fail to generalize to unseen or noisier data.
* The CICIDS2017 dataset is well-structured and clean, limiting real-world complexity.

**Recommendations for Improvement**

Robust Feature Engineering:

* Include feature scaling, encoding protocols/ports, time-series patterns
* Add traffic burstiness, average packet size, etc.

Dataset Diversity:

* Use additional datasets like NSL-KDD or UNSW-NB15
* Combine synthetic and real-world logs for robustness

Regularization and Hyperparameter Tuning:

* Apply grid search or random search for better Random Forest tuning

Model Validation:

* Use k-fold cross-validation
* Validate using entirely different dataset

Ensemble and Comparative Models:

* Try Logistic Regression, Gradient Boost, and SVM

Noise Injection Testing:

* Add label noise, out-of-distribution packets to test real-world applicability

**Summary**

This project demonstrates an end-to-end AI pipeline for detecting cyber threats with strong performance on the CICIDS2017 dataset. However, real-world deployments demand broader preprocessing, validation, and model generalization efforts to ensure security and reliability.