Real-Time Intrusion Detection System Using NSL-KDD and Deep Learning

1. Introduction

This project aims to implement a real-time Intrusion Detection System (IDS) using a deep learning model trained on the NSL-KDD dataset. The system captures live network packets, extracts features, and predicts whether each packet is normal or part of an intrusion.

2. Dataset - NSL-KDD

The NSL-KDD dataset is an improved version of the KDD’99 dataset. It contains labeled records of network connections, categorized into Normal, DoS, Probe, R2L, and U2R attack types. This dataset is widely used for evaluating IDS models.

3. Feature Selection and Preprocessing

From the 41 available features, a subset of 5 was selected: protocol_type, service, flag, src_bytes, and dst_bytes. Additionally, two engineered features were included: ttl (time to live) and pkt_len (estimated packet length). Categorical features and labels were label-encoded, and all features were standardized using StandardScaler. LabelEncoder and Scaler were saved using joblib for reuse during prediction.

4. Model Architecture

A sequential deep learning model was created using Keras. It contains: - Input layer with 7 features - Dense layer with 64 units (ReLU) - Dense layer with 32 units (ReLU) - Output layer with softmax activation The model uses sparse categorical crossentropy as loss and Adam optimizer. It was trained over 10 epochs with a batch size of 32 and saved as 'ids_dnn_model_7features.h5'.

5. Real-time Packet Capture and Prediction

Using Scapy, the system captures live IP packets. From each packet, 7 features are extracted in the same format as the training data. These features are scaled and fed into the trained model. The predicted label is decoded and displayed with the source IP and timestamp. To reduce duplicate outputs, processed IPs are cached and cleared periodically.

6. Output Example

Example output: Timestamp: 2025-08-07 15:05:00 | Source IP: 192.168.1.85 | Prediction: normal

7. Saved Files

- ids_dnn_model_7features.h5: Trained Keras model - scaler.pkl: StandardScaler for preprocessing - label_encoder.pkl: LabelEncoder for decoding predictions

8. Advantages

- Real-time prediction - Lightweight feature set - Deep learning-based accuracy - Modular and extensible system

9. Future Improvements

- Expand to full 41 features - Integrate alerting or blocking mechanisms - Visualize traffic predictions on dashboard - Use autoencoders or LSTM for anomaly detection

10. Conclusion

This project presents a compact yet effective IDS built on deep learning and live packet capture. It lays the foundation for more advanced network security tools.