Project Proposal

Network Intrusion Detection Agent

Introduction

This project proposes the design and implementation of an intelligent learning agent for network intrusion detection. The agent's goal is to analyze network traffic flows and autonomously decide whether a given flow is benign or malicious, and if malicious, identify the type of cyber attack.

The system is designed as a two-stage intelligent agent:

- **Stage 1**: Binary Detection: Classifies network traffic as Benign or Attack.
- **Stage 2**: Attack Classification: If an attack is detected, the agent further classifies it into a specific attack category (DDoS, DoS, scanning, injection, etc).

Dataset Explanation

The project will use the NF-UQ-NIDS-v2 Network Intrusion Detection Dataset, a modern, large-scale network intrusion detection dataset publicly available on Kaggle. This dataset was chosen because:

- It represents realistic network traffic collected from real environments.
- It contains both benign and multiple attack types, enabling hierarchical classification.
- It includes over 30 features related to packet counts, byte statistics, protocol information, and timing characteristics.
- It reflects class imbalance, a common and realistic challenge in cybersecurity systems.

The dataset contains tens of millions of rows, which motivates the use of sampling and efficient preprocessing strategies, a design decision that will be discussed in the final report.

Agent Type

The proposed system is a learning agent with the following characteristics:

- **Learning component:** Supervised machine learning models trained on labeled network traffic.
- **Performance element:** Uses trained classifiers to make decisions on unseen network flows.
- **Critic:** Performance is evaluated using quantitative metrics such as Precision, Recall, F1-score, and Accuracy.
- **Environment:** A dataset representing network traffic behavior.

Environment Description

The agent operates in a partially observable, static, discrete environment:

- **Partially observable:** The agent only sees summarized flow-level features, not raw packets.
- **Static:** Each network flow is classified independently.
- **Discrete:** Outputs are discrete labels (Benign / Attack types).
- **Single-agent:** The agent operates independently without coordination.

AI Techniques

The project will integrate multiple AI techniques:

Technique 1: Statistical Learning (Random Forest, XGBoost)

- **Random Forest** will be used as a baseline due to its robustness, interpretability, and resistance to overfitting.
- **XGBoost** will be used as an advanced ensemble method, known for high performance on structured tabular data.

Using both allows comparative evaluation between them and discussion of trade-offs in model complexity, accuracy, and computational cost.

Technique 2: Hierarchical Decision Structure (Two-Stage Reasoning)

Rather than treating the task as a single flat multi-class problem, the agent will reason in two stages:

1. Decide whether a threat exists.
2. Decide what type of threat it is.