Traffic Anomaly Detection in SDN Using Deep Learning

Motivation:

- Rule-based firewalls are rigid and outdated
- SDN offers control, but opens new attack risks
- Deep Learning enables smart, real-time anomaly detection

Key Points:

- Simulate SDN using Mininet and Floodlight
- Collect flow-level traffic data using Floodlight Rest APIs
- Use DL models such as LSTM/Transformers to detect unusual traffic behaviour in real time

Simulation Topology & Floodlight Controller

Network Setup:

- 6 Hosts (h1–h6)
- 3 Switches (s1–s3) in tree topology (depth=2, fanout=2)
- 1 Floodlight Controller running in Docker

Traffic:

- Normal: ping, iperf, curl
- Attacks: hping3, scapy, port scanning scripts

System Modules & Data Flow

Floodlight Controller	Runs SDN logic, collects flow stats
REST API	Provides access to flow-level data
Traffic Generator	Scripts that simulate normal & malicious traffic
Feature Extractor	Parses flow stats (packet/byte counts, ports, duration)
ML Model (LSTM/TF)	Classifies time-series as normal or anomalous
Response System	Logs or blocks flows based on ML decision

Anomaly Detection Using Deep Learning

Binary classification (sequence-to-label)

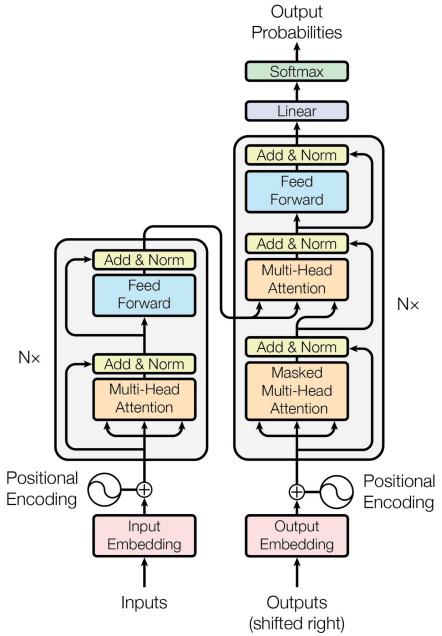
- Input: Sequence of flow stats (e.g., 10 time steps)
- Target: 0 = Normal, 1 = Anomaly

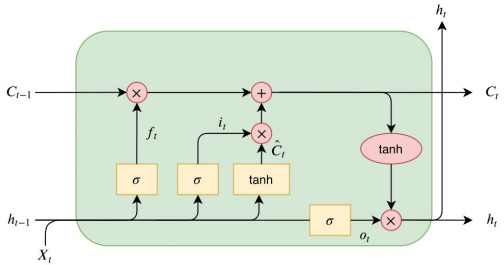
Selected Features:

- packet_count, byte_count, duration_sec
- Derived: packet_rate, byte_rate, avg_packet_size
- Others: src_port, dst_port, protocol, connection_rate

Labeling:

- Normal traffic: 0
- DoS/Scan traffic: 1 (based on scenario control)





Results and Value of the Project

Expected Results:

- High accuracy on simulated traffic
- Detects DoS, port scanning, burst traffic using temporal patterns
- Works in near real-time by continuously polling the controller