

Anomaly-Based Network Intrusion Detection Using One-Class Classification

Controlled benchmark of six one-class classifiers across two public intrusion datasets

Mohamed Amine Chamam • Capstone Project, MedTech • Supervisor: Dr. Ikram Saadaoui • Defended July 2026 — Grade: Outstanding • IEEE conference paper: submitted, under review

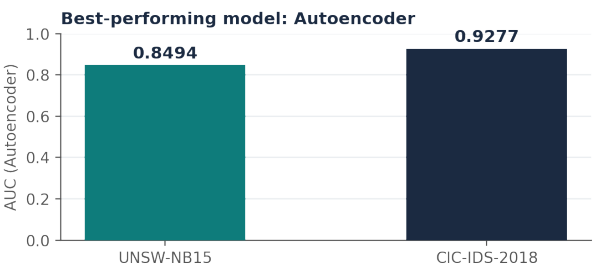
Problem

Network intrusion detection systems rarely have labeled attack data to train on, and new attack types are unseen by definition. This project evaluates whether one-class classifiers — trained only on normal traffic — can reliably flag anomalous (potentially malicious) traffic without ever seeing an attack example during training.

Approach

- Implemented and benchmarked **six one-class models** under one controlled protocol: Autoencoder and Deep SVDD (built from scratch in NumPy/PyTorch), Isolation Forest, One-Class SVM, PCA Reconstruction Error, and Minimum Covariance Determinant.
- Evaluated on two large public benchmarks: **UNSW-NB15** and **CIC-IDS-2018**.
- Built a **leakage-free evaluation pipeline**: all preprocessing (log1p transform, frequency encoding, quantile clipping) fit exclusively on normal training data; decision thresholds selected via Youden's J statistic on a held-out validation split — never on the test set.

Results



- The **Autoencoder** performed best overall, reaching an AUC of **0.8494** on UNSW-NB15 and **0.9277** on CIC-IDS-2018.
- Deep SVDD** showed partial hypersphere collapse on CIC-IDS-2018: a conservative failure mode (low false-positive rate, high false-negative rate).
- A score-level **ensemble of all six models underperformed** the single best model by 4.5–17.4 AUC points, showing that naive combination hurts more than it helps here.

Why It Matters

Demonstrates rigorous experimental methodology (no data leakage, reproducible thresholding) alongside practical ML engineering: from-scratch implementation of two deep models, systematic benchmarking across datasets, and honest reporting of failure modes rather than cherry-picked results.

Skills

Python	PyTorch	NumPy	scikit-learn	Anomaly Detection
Deep Learning	Model Evaluation	Cybersecurity	Reproducible ML	

Full code, from-scratch model implementations, and detailed methodology available on request or via GitHub. The associated IEEE conference paper is currently under review.