Proactive Privacy-Preserving Defense for Federated Learning based Intrusion Detection System

With Homomorphic encryption and unlearning mechanism

GUIDED BY:

Dr. K. Kulothungan

Associate Professor,

Department of Information Science and Technology

LOGESH M S 2022115045	VIJAY K G 2022115063	JOTHIRUBAN M 2022115096	SYED SHARAAFATH HASSAN S 20221151030

Overview

- A secure way for Clients (Institutes) to learn together without sharing private data for training an Intrusion Detection System.
- The System uses advanced encryption to protect information and actively stops hackers from corrupting the model.
- It also lets us 'delete' specific user data when needed and accurately spots network attacks, ensuring a safe and reliable network for everyone.

Literature Survey

S no	Title of the Paper	Author and Year	Methods
1	A Proactive Defense Against Model Poisoning Attacks in Federated Learning (RECESS)	Zhang et al. (2025) IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING,VOL.22	Introduces "RECESS," a proactive defense framework. The server sends "Query Gradients" (Trap Vectors) to clients and verifies their response consistency using Cosine Similarity
2	FedPHE: A Secure and Efficient Federated Learning via Packed Homomorphic Encryption	Li et al. (2025) IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING,VOL.22	Uses CKKS Homomorphic Encryption with SIMD Packing to encrypt multiple gradients into a single vector. Implements a "Batching" technique to reduce ciphertext size.

Literature Survey

S no	Title of the Paper	Author and Year	Methods
3	FedRecovery: Differentially Private Machine Unlearning for Federated Learning Frameworks	Zhang et al. (2023) IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 18	Develops a "Machine Unlearning" algorithm. It tracks historical updates and allows the server to mathematically subtract the influence of a specific client from the global model..
4	Evaluating Federated Learning-Based Intrusion Detection Scheme for Next Generation Networks	Singh et al. (2024) IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, VOL. 21,	Proposes a Federated IDS using deep learning on the CIC-IDS2017 dataset. Focuses on handling Class Imbalance (rare attacks vs normal traffic).

Problem Statement

- Collaborative organizations fighting cyber threats face regulatory restrictions (e.g., **GDPR**).
- Standard Federated Learning (FL) **avoids raw data sharing** but risks "Gradient Leakage," allowing mathematical reconstruction of sensitive data from shared model updates.
- FL is also highly susceptible to Model Poisoning, where **malicious data corrupts the global AI**, which is hard to detect since the central server cannot inspect updates.
- Existing defenses are reactive, blocking future attacks but unable to "unlearn" or **reverse specific malicious contributions** efficiently without complete model retraining.

Challenges

- Implementing high-security encryption on AI models increases computation time significantly. The challenge is **achieving Real-Time Intrusion Detection speeds** while processing complex, encrypted vectors.
- The system must distinguish between "Benign" and "Malicious" updates without ever seeing the data. This requires **implementing complex Similarity Checks** that operate blindly on encrypted inputs.
- **Real-world datasets (CIC-IDS2017)** contain less than 1% attack traffic. Training a decentralized model to detect these rare events without producing high false positives requires specialized Local Data Sampling techniques.

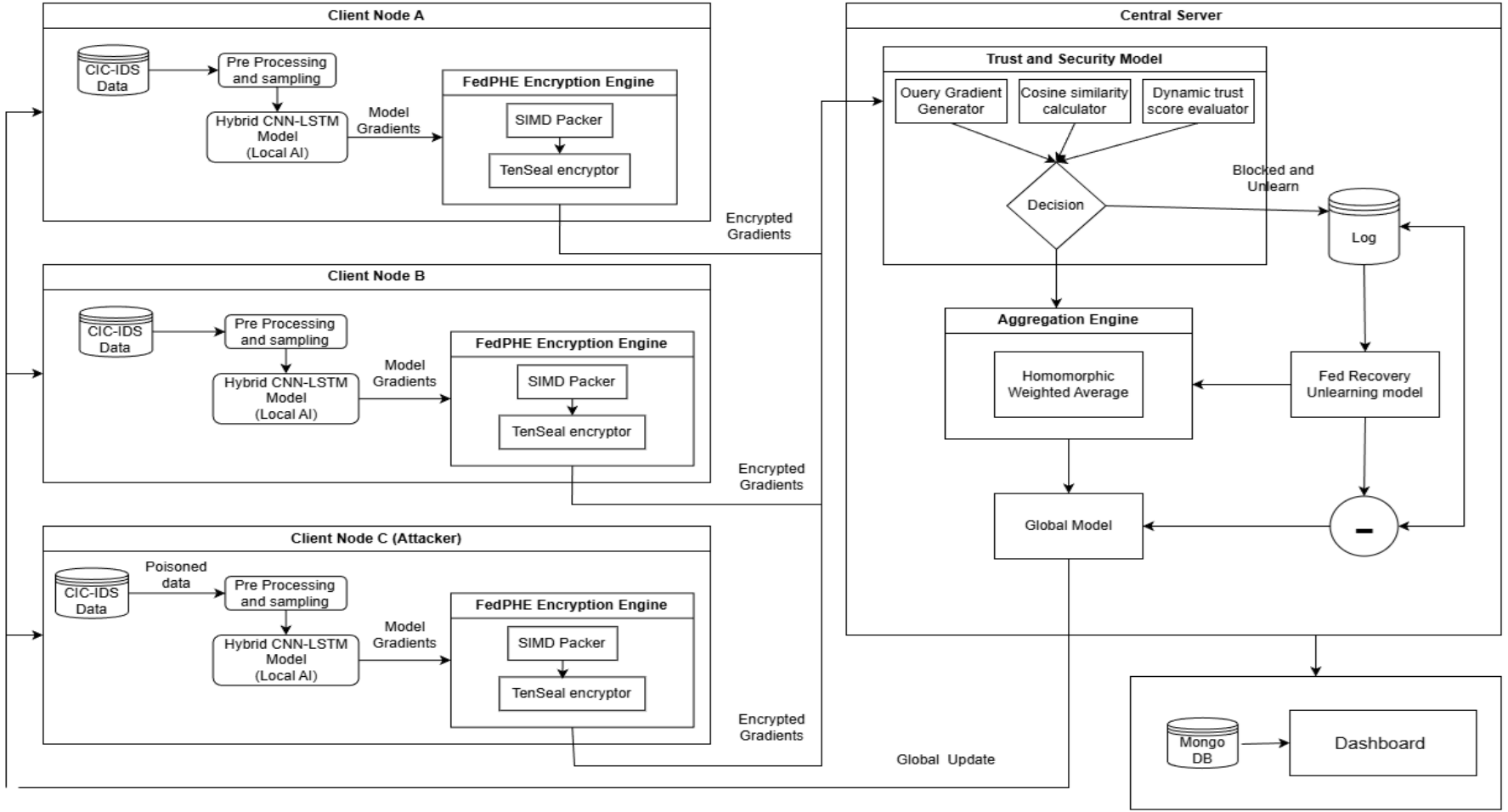
Objectives

- To build a Hybrid CNN-LSTM Neural Network that effectively detects modern cyberattacks (like DDoS and Botnets) across distributed clients.
- To integrate Packed Homomorphic Encryption (FedPHE) ensuring that the central server aggregates model updates without ever accessing the raw gradients or client data.
- To engineer a server-side defense module that uses "Trap Gradients" and encrypted similarity checks to identify and block malicious nodes before their updates are fully merged.
- To implement a remediation module that allows the system to subtract the encrypted contributions of a detected attacker from the global model, ensuring the AI can recover itself from past poisoning.

Scope

- Simulates a collaborative intrusion detection network using the CIC-IDS2017 dataset to detect anomalies like DDoS and Botnets across disparate, independent client nodes.
- Implements Packed Homomorphic Encryption (FedPHE) to ensure Blind Aggregation allowing the server to combine model updates without ever decrypting or seeing the underlying values.
- Specifically addresses Data Poisoning Attacks (Label Flipping) where compromised internal nodes attempt to degrade the global model's accuracy.
- Develops a full-stack platform featuring a Hybrid CNN-LSTM classification model, a Unlearning Module, and a Real-time Trust Dashboard for monitoring node integrity.

Architecture Diagram


Client Node A
CIC-IDS Data
Pre Processing and sampling
Hybrid CNN-LSTM Model (Local AI)
Model Gradients
FedPHE Encryption Engine
SIMD Packer
TenSeal encryptor
Encrypted Gradients
Client Node B
CIC-IDS Data
Pre Processing and sampling
Hybrid CNN-LSTM Model (Local AI)
Model Gradients
FedPHE Encryption Engine
SIMD Packer
TenSeal encryptor
Encrypted Gradients
Client Node C (Attacker)
CIC-IDS Data
Poisoned data
Pre Processing and sampling
Hybrid CNN-LSTM Model (Local AI)
Model Gradients
FedPHE Encryption Engine
SIMD Packer
TenSeal encryptor
Encrypted Gradients
Global Update
Central Server
Trust and Security Model
Ouery Gradient Generator
Cosine similarity calculator
Dynamic trust score evaluator
Decision
Blocked and Unlearn
Log
Aggregation Engine
Homomorphic Weighted Average
Fed Recovery Unlearning model
Global Model
Mongo DB
Dashboard

TOOLS & TECHNOLOGIES

- **Python:** The primary language used across all modules for implementing the FL server, client logic, and cryptographic protocols.
- **PyTorch:** The unified framework used in all the papers for building and training the neural networks.
- **Torchvision:** Essential for loading and preprocessing the image datasets mentioned in the studies
- **TenSEAL:** A library for Homomorphic Encryption operations on tensors

- **TensorBoard:** Used for tracking model convergence, visualizing loss curves, and monitoring training metrics in real-time.
- **CUDA Toolkit:** Required for enabling GPU acceleration on NVIDIA hardware
- **Matplotlib:** Recommended for generating the static performance graphs
- **NumPy:** Essential for matrix manipulations

REFERENCES

- G. Singh, K. Sood, P. Rajalakshmi, D. D. N. Nguyen and Y. Xiang, "Evaluating Federated Learning-Based Intrusion Detection Scheme for Next Generation Networks," in IEEE Transactions on Network and Service Management, vol. 21, no. 4, pp. 4816-4829, Aug. 2024
- Y. Li et al., "FedPHE: A Secure and Efficient Federated Learning via Packed Homomorphic Encryption," in IEEE Transactions on Dependable and Secure Computing, vol. 22, no. 5, pp. 5448-5463, Sept.-Oct. 2025
- H. Yan et al., "A Proactive Defense Against Model Poisoning Attacks in Federated Learning," in IEEE Transactions on Dependable and Secure Computing, vol. 22, no. 4, pp. 3529-3543, July-Aug. 2025
- L. Zhang, T. Zhu, H. Zhang, P. Xiong and W. Zhou, "FedRecovery: Differentially Private Machine Unlearning for Federated Learning Frameworks," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 4732-4746, 2023

THANK YOU