DATA-PRIVACY FOCUSED FEDERATED LEARNING FRAMEWORK FOR INDUSTRIAL IOT

R25-039

Supervisor: Mr. Amila Nuwan Senerathne
Co-Supervisor: Tharaniwarma Kumaralingam

The dissertation was submitted in partial fulfilment of the requirements for the B.Sc. (Honors) degree in Information Technology

Department of Computer Systems and Engineering

Sri Lanka Institute of Information Technology
Sri Lanka

August 2025

Declaration

We declare that this is our own work, and this dissertation does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text. Also, I hereby grant to Sri Lanka Institute of Information Technology the non-exclusive right to reproduce and distribute my dissertation in whole or part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

	Member Name	Registration Number	Signature
1.	Nanayakkara Y.D.T.D	IT21826368	
2.	Mendis H.R.M	IT21822612	
3.	Weerasinghe K.M	IT21831904	
4.	Dissanayaka K.D.A.R. A	IT21828348	

	Title	First Name	Last Name	Signature
Supervisor	Mr.	Amila	Senadheera	
Co-Supervisor	Mr.	Tharaniyawarma	Kumaralingam	
External Supervisor	-	-	-	-

Summary of external supervisor's (if any) experience and expertise

ABSTRACT

This thesis presents and tests an integrated, modular framework for secure, privacy-preserving, and robust federated learning geared to Industrial IoT contexts. Addressing gaps in traditional FL approaches, the system unites secure communication, protocol enforcement, differential privacy, homomorphic encryption, dropout-robust aggregation, and real-time adversarial defense into a practical platform for large-scale, heterogeneous, and resource-constrained deployments. The Secure Communication and Protocol Enforcement Module ensure end-to-end confidentiality, mutual authentication, message integrity, and strict access control for all command-and-control and model-update channels, protecting distributed learning pipelines from unauthorized access, manipulation, and injection attacks. Privacy is retained via customizable Differential Privacy and Homomorphic Encryption, offering high secrecy with tunable trade-offs for industrial applications. The Secure Aggregation Module's mix of pairwise masking and threshold secret sharing provides aggregation that is robust to client dropout and hides individual contributions, while rigorous empirical evaluation demonstrates that global model accuracy is not sacrificed.

Real-time detection and mitigation of poisoning, Byzantine, and backdoor threats are achieved by a lightweight anomaly detection pipeline, robust aggregation (e.g., Krum), and adaptive response anchored in round-by-round client trust scoring and rollback. Across simulated IIoT settings under adversarial stress and client churn, the framework reliably blocks or rejects malicious behavior, maintains model correctness, and provides privacy even in the face of persistent attacks and inconsistent connectivity. Histogram and success rate analysis reveal indistinguishability of outputs under user removal and near-theoretical robustness to dropout until cryptographic limitations are reached. The complete system is constructed and tested in a multi-layered architecture using industry-standard security libraries, efficient neural network pipelines, and scalable orchestration tools appropriate for open-source release and real-world industrial pilot deployments.

Collectively, these studies extend the state-of-the-art in trustworthy distributed machine learning by proving that attack-resistant, privacy-guaranteed, and operationally robust federated learning is attainable for mission-critical industrial contexts. This study creates a basis for secure, compliant, and adaptive analytics across the IIoT, and provides a plan for expanding federated learning's role in future cyber-physical infrastructure.

Table of Contents

LIST OF FIGURES

List of abbreviations

- IIoT: Industrial Internet of Things
- FL: Federated Learning
- TLS: Transport Layer Security
- SCPM: Secure Communication and Protocol Enforcement Module
- HMAC: Hash-based Message Authentication Code
- RBAC: Role-Based Access Control
- DP: Differential Privacy
- HE: Homomorphic Encryption
- PPM: Privacy Preservation Module
- SAM: Secure Aggregation Module
- ADRM: Attack Detection and Resilience Module
- ML: Machine Learning
- OT: Operational Technology
- IID: Independent and Identically Distributed (data)
- APT: Advanced Persistent Threat
- DDoS: Distributed Denial of Service
- AI: Artificial Intelligence
- IoT: Internet of Things

1. INTRODUCTION

The Industrial Internet of Things (IIoT) integrates diverse sensors, actuators, and control systems to facilitate real-time monitoring, predictive maintenance, and automated decision-making; however, this connectivity generates sensitive operational data and broadens the cyber attack surface, necessitating stringent assurances of confidentiality, integrity, and availability at an industrial scale. Federated Learning (FL) integrates seamlessly with IIoT by retaining raw data locally and transmitting only model updates for global training, alleviating bandwidth limitations and minimizing central data exposure compared to conventional centralized learning, while ensuring ongoing adaptation at the edge. Nonetheless, practical federated learning implementations are vulnerable throughout the pipeline: controlling channels may be intercepted or spoofed, model updates can expose private information through inversion or gradient attacks, simplistic aggregation may disclose individual contributions or falter due to client dropouts, and adversarial actions such as poisoning or Byzantine faults can compromise model integrity in decentralized, resource-limited settings.

This thesis proposes a comprehensive, modular framework to rectify systemic deficiencies by securing communication, preserving privacy, enhancing aggregation, and protecting against adversaries throughout the federated learning lifecycle in Industrial Internet of Things environments, thereby providing defense-in-depth architecture tailored for diverse devices and unreliable networks. First, the Secure Communication and Protocol Enforcement Module protect the FL controls system with TLS 1.3 and mutual authentication for encrypted, identity-verified sessions, HMAC for message integrity, and Role-Based Access Control (RBAC) to constrain sensitive actions to authorized roles mitigating man-in-the-middle, replay, and unauthorized command injection across synchronization and model-exchange workflows. The Privacy Preservation Module enhances Federated Learning by incorporating Differential Privacy and variants of Homomorphic Encryption to limit information leakage from client updates and, when necessary, facilitate encrypted aggregation, alongside an empirical analysis of the accuracy-efficiency-privacy trade-offs relevant for industrial applications. The Secure Aggregation Module employs pairwise masking and Shamir’s Secret Sharing to guarantee that the aggregator obtains solely the aggregate, accommodates client dropouts, and maintains both individual update confidentiality and training advancement in unreliable networks, all without compromising the final model accuracy compared to non-secure baselines. The Attack Defense and Resilience Module offer efficient anomaly detection and specific protection against model poisoning, Byzantine behavior, and associated adversarial strategies, therefore enhancing detection precision and operational stability within limited computational resources and bandwidth.

Collectively, these elements achieve four cohesive goals for IIoT-grade federated learning: (i) secure and authenticated command and control with integrity-verified messaging and detailed authorization; (ii) privacy-preserving learning through differential privacy and homomorphic encryption with estimated overheads; (iii) privacy-preserving, dropout-resilient aggregation that conceals individual updates; and (iv) real-time attack detection and mitigation that maintains global model quality in adversarial environments. The resulting contributions include a deployable C2 security architecture tailored to FL in IIoT, a comparative privacy study across FL, FL+DP, FL+HE, and FL+DP+HE configurations, a practical secure aggregation protocol validated against baseline accuracy while tolerating client failures, and an ADR pipeline that enhances resilience with low computational overhead for edge-constrained devices, thereby advancing a scalable, privacy-aware, and robust federated learning framework for critical industrial operations.

1.1 Literature Review

Federated Learning can improve security, privacy, and resilience in Industrial Internet of Things (IIoT) environments, according to the literature review. The primary goal of early FL research was to lower communication costs in distributed learning systems, which is crucial in IIoT networks with latency and bandwidth constraints. Establishing the foundation for more sophisticated privacy and security measures, foundational works showed FL's potential for privacy-aware distributed model training.

As FL developed, the emphasis shifted from efficiency to protecting against advanced cyberthreats such gradient inversion attacks, Byzantine behaviors, and data and model poisoning. Malicious clients can contaminate global models, adversaries can use gradients to reconstruct private data, and intricate protection mechanisms can use up all available processing power in collaborative learning. Several strong aggregation algorithms, such Trim, Mean and Krum, have been created to filter out fraudulent updates to lessen these difficulties. To guarantee the confidentiality of individual client contributions, secure aggregation protocols and hybrid cryptographic techniques have been presented concurrently [1].

The preservation of privacy with Differential Privacy (DP) and Homomorphic Encryption (HE) has been the subject of another important research area. By adding calibrated noise to gradients, DP reduces the possibility of leakage or inversion and protects model updates [2]. Contrarily, HE maintains confidentiality throughout the training process by enabling the aggregation of encrypted updates without ever disclosing the underlying data to the server. Both approaches, however, present difficulties in IIoT environments, especially with relation to communication and processing overhead. According to studies, HE provides more privacy at a larger expense, while DP provides useful protection with little loss of utility. To balance these trade-offs, hybrid DP-HE techniques are becoming more popular; nonetheless, they need to be carefully adjusted for IIoT deployments with limited resources [3].

The literature emphasizes the increasing focus on defense and resilience in decentralized learning systems, in addition to aggregation and privacy. Due to its distributed architecture, the IIoT is more vulnerable to attacks such as DDoS, data poisoning, and Byzantine manipulation [1]. Secure aggregation is inadequate against more adaptive adversaries, even while it aids in hiding malicious updates. To recognize and react to anomalous contributions in real time, sophisticated techniques are required, such as adaptive resilient architectures and anomaly detection. As a decentralized trust and auditability mechanism, blockchain-based FL has also been proposed [4]; however, many of these solutions are either unproven or too resource-intensive for extensive IIoT contexts. Therefore, creating a cohesive and lightweight security system that can handle changing threats while still being effective for a variety of edge devices with low resources is the current problem.

Another crucial aspect of IIoT-FL research is communication and protocol-level security. Major vulnerabilities exist in legacy and unencrypted protocols used in industrial systems, which could allow model updates to be intercepted or manipulated. TLS 1.3 is the recommended standard, according to recent studies, because of its stronger encryption, forward secrecy, and mutual authentication features, which are essential for thwarting impersonation and man-in-the-middle attacks. However, encryption is not enough on its own; in order to stop unwanted operations or backdoor injections, stringent role-based access control (RBAC) [5], secure command-and-control channels, and message integrity checks via HMAC are necessary.

Although separate elements like secure aggregation, privacy-preserving computation, and encrypted communication each address particular facets of FL security, there is still a substantial gap in their integration into a coherent, scalable, and effective framework appropriate for actual IIoT deployment, according to the literature currently in publication. The research frontier in secure federated learning for industrial systems is still defined by the requirement for a comprehensive defense plan that strikes a compromise between privacy, performance, and resilience.

1.2 Research Gap

Most Despite considerable progress in federated learning (FL) for Industrial IoT (IIoT), notable research gaps persist in establishing a safe, privacy-preserving, and resilient framework that functions effectively in diverse and resource-constrained settings.

A significant gap exists in secure aggregation. Although various protocols such as pairwise masking, homomorphic encryption, and secret sharing have been suggested to safeguard client updates, the majority presume consistent client capabilities and stable connectivity [6], which rarely align with industrial realities. Conventional methods frequently burden limited devices with excessive computing and bandwidth requirements, rendering them inappropriate for extensive IIoT implementations. Moreover, strong aggregation procedures intended to eliminate malicious updates often diminish accuracy in non-IID and irregular data distributions characteristic of industrial systems [7]. Current protocols provide fixed security settings, failing to dynamically adjust privacy, model accuracy, and efficiency in response to varying network and device conditions. Therefore, the critical task is to develop a dynamic, resource-efficient, and scalable safe aggregation technique that accommodates the conditions of IIoT environments marked by dropouts, slower nodes, and adverse disruptions.

Also, a significant concern is the privacy preservation gap, wherein the challenge lies in upholding robust confidentiality without sacrificing performance. Techniques like as Differential Privacy (DP) and Homomorphic Encryption (HE) provide substantial protection but include trade-offs that are especially onerous for industrial devices with constrained computing power and communication capabilities [8]. Although differential privacy protects model updates by noise addition and homomorphic encryption facilitates computations on encrypted data, their combination frequently leads to excessive complexity. Most present research ignore solutions for dynamically modifying privacy budgets or encryption parameters in response to the different capabilities and compliance requirements of industrial stakeholders. Therefore, the research needs adaptive privacy-preserving algorithms that intelligently balance privacy assurances with computing practicality and accuracy in real-world IIoT installations.

The defense and resilience aspects represent another open research area. Current defense techniques in FL primarily handle isolated threats such as model poisoning, Byzantine behavior, or label-flipping but fail to provide comprehensive protection over the full adversarial spectrum. Many of these technologies entail significant runtime or communication cost, limiting their scalability in distributed industrial environments [9]. Few solutions incorporate real-time anomaly detection, robust aggregation, and automatic countermeasures (e.g., quarantine or rollback) into a cohesive, lightweight pipeline. This underlines the necessity for an integrated, real-time security system capable of detecting, isolating, and mitigating coordinated or evolving threats rapidly, while ensuring operational continuity under IIoT' s resource and connectivity constraints.

Finally, considerable gaps persist in secure communication and protocol enforcement for federated learning in industrial networks. Many frameworks continue to use on old or insufficiently protected protocols, such as TLS 1.2, enabling limited mutual authentication and weak protection against replay, man-in-the-middle, or injection attacks. Even where encryption is applied, weak identity verification, authorization, and coarse-grained access controls allow compromised devices to impact the global model. In addition, key management and certificate distribution sometimes struggle to scale successfully across thousands of industrial nodes. The absence of fine-grained, role-based command enforcement and unified cross-layer protection combining encryption, authentication, message integrity, and governance represent a fundamental shortcoming. Therefore, the primary research gap in this sector is the development of a fully integrated, lightweight, and scalable protocol enforcement architecture that guarantees end-to-end security across all communication layers without overburdening IIoT devices.

In summary, the research consistently demonstrates that while individual security, privacy, and defense components of federated learning have grown separately, a unified, adaptive, and resource-efficient architecture suitable for the IIoT environment remains missing. Addressing this broad gap is critical to enable trustworthy, large-scale, and realistic industrial federated learning deployments that can survive real-world adversarial, operational, and infrastructure obstacles.

1.3 Research problem

Industrial IoT creates sensitive, high-volume data across heterogeneous, resource-constrained devices, where centralized learning is problematic and FL, while promising, presents additional attack surfaces in communication, model update privacy, aggregation, and adversarial manipulation during training. Existing point solutions address isolated risks but often introduce heavy costs or fail under real-world constraints like client dropouts, non-IID data, and unreliable connectivity, leaving an unmet need for a cohesive, deployable framework that balances security, privacy, robustness, and efficiency at scale.

- Secure Communication and Protocol Enforcement: Secure IIoT–FL control traffic against MitM, replay, and unauthorized commands by building a low-overhead layer with TLS 1.3 mutual authentication, HMAC integrity, and RBAC that works on constrained devices.
- Privacy Preservation Module: Limit information leakage from model updates using tunable Differential Privacy and Homomorphic Encryption, balancing privacy budgets and ciphertext costs with acceptable accuracy and bandwidth for industrial workloads.
- Secure Aggregation Module: Ensure the server learns only the aggregate and training survives client dropouts by combining pairwise masking with Shamir's Secret Sharing, while adapting mask precision to heterogeneous IIoT devices without hurting baseline accuracy.
- Attack Defense and Resilience Module: Detect, isolate, and recover from poisoning and Byzantine attacks in real time via a lightweight anomaly-detection pipeline integrated with robust aggregation, keeping latency low and model quality high at scale

1.4 Research Objectives

1.4.1 Main objective

Create, put into practice, and assess an integrated IIoT federated learning framework that offers low overhead real-time attack defense across diverse, resource-constrained devices and unreliable networks while also protecting update privacy, guaranteeing confidential and dropout-robust aggregation, and securing command-and-control traffic.

1.4.2 Specific Objectives

1. **Secure Aggregation.**

- Design and implement a multi-round, resource-aware secure aggregation protocol adapted to IIoT limitations, supporting scalable and fault-tolerant model aggregation.
- Integrate pairwise masking and threshold cryptography (e.g., Shamir's Secret Sharing) to protect client updates and enable aggregate recovery in the event of client dropouts.
- Evaluate the protocol's correctness, computational efficiency, and accuracy under typical IIoT network scenarios, including unreliable connectivity and hostile environments.

2. **Privacy Preservation**

- Incorporate Differential Privacy (DP) and Homomorphic Encryption (HE) techniques into federated learning workflows to enhance confidentiality of both local and global model updates.
- Quantify the trade-offs between privacy, accuracy, computation, and communication overhead under several configurations (simple FL, FL+DP, FL+HE, and FL+DP+HE).
- Develop guidelines for tuning privacy budgets and encryption parameters to balance privacy protection, model performance, and compliance with industrial data governance regulations.

3. **Attack Defense and Resilience**

- Engineer a lightweight, real-time security system to identify, isolate, and mitigate adversarial behaviors such as poisoning, Byzantine attacks, and label-flipping.
- Integrate anomaly detection models and automated mitigation techniques (e.g., down-weighting, quarantine, rollback) to ensure global model integrity.
- Ensure the defense module runs efficiently on resource-limited edge devices, supporting scalability and minimal latency throughout distributed learning cycles.
- Evaluate defense efficacy by precision, recall, time-to-response, and influence on model integrity in simulated IIoT hostile scenarios.

4. **Secure Communication and Protocol Enforcement**

- Develop a secure command-and-control (C2) communication layer employing TLS 1.3 with mutual authentication to assure secrecy and authenticity in all data exchanges.
- Implement HMAC-based integrity verification and role-based access control (RBAC) to enforce message authenticity, prevent tampering, and limit rights to authorized entities.

- Test and assess the communication layer’s security, scalability, and performance across actual IIoT deployments, emphasizing minimal overhead and solid operational assurances.

The final result is a modular, open-source federated learning platform that enables end-to-end privacy, security, and resilience for IIoT contexts. The system will exhibit great scalability, robustness against adversarial attacks, secure communication between devices, and adaptive performance on heterogeneous industrial networks enabling trustworthy, large-scale industrial machine learning adoption.

2 PROJECT REQUIREMENTS

2.1 Functional Requirements

This section discusses the critical capabilities and behaviors necessary for the integrated Federated Learning (FL) framework for Industrial IoT (IIoT) environments. The following functional criteria collectively guarantee secure, privacy-preserving, and resilient federated learning operations across heterogeneous devices and networks. They handle crucial issues such as communication security, access control, privacy enhancement, secure aggregation, anomaly detection, and system monitoring enabling dependable and trustworthy model training within limited industrial ecosystems.

- **Secure Communication**
  The system shall protect all FL control and model-update channels using TLS 1.3 with mutual authentication, ensuring encrypted, identity-verified communication between IIoT clients and edge servers.

- **Message Integrity and Freshness**
  The system shall guarantee per-message integrity using HMAC and provide nonce/timestamp freshness to avoid tampering and replay attacks across all control and model-update messages.

- **Access Control and Auditability**
  The system shall implement role-based access control (RBAC) over sensitive operations such as client enrollment, aggregation initiation, model release, and policy modification and keep auditable logs for all authorization decisions.

- **Certificate Lifecycle Management**
  The system shall provide certificate creation, rotation, and revocation to maintain up-to-date trust anchors for clients and servers throughout large-scale IIoT deployments.

- **Privacy-Preserving Learning Options**
  The system shall support Differential Privacy (DP) on client updates with customizable privacy budgets and gradient clipping, and Homomorphic Encryption (HE) as an optional encrypted aggregation mode.It shall provide policy-driven selection among FL, FL+DP, FL+HE, and FL+DP+HE modes based on data sensitivity, device capability, and network conditions.

- **Secure Aggregation and Masking**
  The system shall disguise individual client updates by pairwise masking such that only aggregate values are revealed to the server.It shall withstand client dropouts using Shamir's

Secret Sharing (SSS) for threshold-based mask recovery, assuring aggregation correctness without disclosing any single update.The system shall offer resource-aware masking precision to adapt compute and communication costs to varied device capabilities while retaining privacy guarantees.

- **Robust Aggregation and Anomaly Detection**
  The system shall assess incoming updates in real time utilizing lightweight anomaly detection features (e.g., gradient norms, similarity, and variance analysis). It shall integrate robust aggregation methods capable of automatically down-weighting, rejecting, or quarantining malicious or aberrant updates to combat poisoning and Byzantine attacks with minimal delay.

- **Recovery and Continuity Mechanisms**
  The system shall feature rollback and checkpoint recovery capabilities and support adaptive retraining to restore model integrity following detected anomalies or attacks.

- **Metrics and Monitoring**
  The system shall measure and provide privacy–utility–efficiency metrics (accuracy, bandwidth, latency, CPU/memory), detection performance (precision, recall, time-to-mitigation), and the influence on model convergence under actual IIoT scenarios.

- **Configuration Management and Optimization**
  The system must offer an integrated configuration interface to co-tune DP noise levels, HE parameters, secure aggregation thresholds, and anomaly-detection sensitivity to fulfill accuracy, privacy, and latency requirements in dynamic IIoT networks.

- **Federation Membership Management**
  The system shall offer lifecycle-safe client enrollment and de-enrollment operations consistent with RBAC and certificate policies to maintain a trustworthy federation membership over time.

- **End-to-End Visibility**
  The system shall enable centralized monitoring and reporting of security, privacy, aggregation robustness, and operational parameters to facilitate real-time decision-making in industrial deployments.

The established functional requirements together offer a safe, privacy-preserving, and robust foundation for the proposed Federated Learning framework within Industrial IoT contexts. By integrating advanced cryptographic approaches, robust aggregation, adaptive configuration, and continuous monitoring, the system enables trustworthy model training even under limited, dispersed, and possibly adversarial settings. These functionalities collectively enable dependable collaboration among heterogeneous IIoT devices while keeping strict criteria of confidentiality, integrity, and operational performance necessary for industrial-scale deployments.

2.2 Non-Functional Requirements

The non-functional requirements convert the design objectives of the framework's four fundamental modules secure communication, privacy preservation, secure aggregation, and attack defense into quantifiable system attributes. They guarantee that the proposed Federated Learning framework for Industrial IoT (IIoT) is functionally accurate, secure, privacy-preserving, robust, efficient, and maintainable in actual industrial environments. The specifications consider IIoT limitations, including restricted bandwidth, device diversity, and sporadic connectivity, to ensure reliable large-scale performance.

- **Confidentiality**
  All control and model-update channels must ensure end-to-end encryption and mutual authentication using secure protocols to prevent eavesdropping and impersonation across constrained IIoT clients and edge servers.

- **Message Integrity and Freshness**
  Every control and model message must include tamper-evident integrity verification and replay protection (e.g., HMAC with nonces or timestamps) to maintain trustworthy communication over unreliable links.

- **Least-Privilege and Auditability**
  Sensitive operations shall be governed by role-based access control (RBAC), with immutable audit logs to support traceability and forensic analysis without disrupting normal operations.

- **Certificate Lifecycle Management**
  The platform must support scalable provisioning, rotation, and revocation of digital certificates to sustain trusted identities and maintain secure communication across large IIoT fleets.

- **Privacy Guarantees with Tunability**
  The framework must offer configurable Differential Privacy (DP) parameters such as noise budgets and clipping norms to control information leakage while quantifying utility impacts on industrial datasets.

- **Cryptographic Privacy at Scale**
  Homomorphic Encryption (HE)–based aggregation shall be supported as an optional privacy mode, with parameter tuning to ensure ciphertext size and computational overhead remain practical for edge devices.

- **Aggregation Confidentiality and Dropout Robustness**
  Secure aggregation must ensure the aggregator learns only the aggregate update, never individual contributions, while tolerating client dropouts through threshold recovery mechanisms (e.g., Shamir's Secret Sharing) without interrupting training.

- **Robustness to Attacks**
  The learning process must incorporate real-time anomaly detection and mitigation to defend against poisoning and Byzantine attacks, maintaining reliable convergence with minimal latency overhead.

- **Reliability and Recovery**
  The platform shall support rollback and checkpoint-based recovery, as well as adaptive retraining, to restore model integrity and continuity following detected incidents or failures.

- **Performance and Resource Efficiency**
  Security, privacy, and robustness mechanisms must operate within industrial performance constraints, ensuring acceptable latency, bandwidth use, and CPU/memory consumption on heterogeneous IIoT devices.
- **Observability and Monitoring**
  The platform must provide end-to-end visibility of metrics such as privacy leakage bounds, aggregation correctness under dropouts, anomaly detection accuracy, and resource usage to enable real-time operational insights.
- **Interoperability and Portability**
  Standardized communication protocols, cryptographic libraries, and interfaces should be used to ensure compatibility with diverse IIoT stacks, networks, and hardware architectures.

- **Operability and Maintainability**
- Configuration management, policy updates, and credential handling must be manageable at fleet scale with minimal downtime, well-defined workflows, and secure automation support.

- **Configuration Co-Tuning**

The system must enable coordinated tuning of DP noise levels, HE parameters, secure-aggregation thresholds, and detection sensitivity to meet desired trade-offs among accuracy, privacy, and latency in dynamic industrial networks.

These non-functional requirements make sure that the IIoT Federated Learning framework stays safe, private, strong, efficient, and easy to maintain throughout its life.Each requirement will have measurable acceptance criteria and experimental validation. These will include metrics like accuracy, convergence, latency, bandwidth, CPU/memory usage, energy efficiency, confidentiality, integrity, and detection efficacy in different IIoT situations. These evaluations will help optimize the framework so that it is ready for deployment, making sure that it is trustworthy, available, and performs consistently in different industrial settings.

2.3 System Software Requirements

The following section lists the combined System Software Requirements for the Federated Learning (FL) framework in Industrial IoT (IIoT) settings. These requirements bring together and make sure that the design goals of the four main modules secure communication, privacy preservation, secure aggregation, and attack defense work together to create a single, interoperable software architecture. The requirements listed below outline the basic server, client, middleware, and management features that are necessary to enable secure, privacy-aware, and reliable distributed learning in a limited industrial setting.

- **Federated Learning Server and Clients**
  The platform shall include a federated learning server and lightweight client agents that coordinate training rounds, manage non-IID data, and exchange model updates securely across IIoT networks.
- **Access Control and Audit Logging**
  A role-based access control (RBAC) engine shall protect sensitive operations (client enrollment, aggregation triggers, model release, and policy changes) and maintain immutable audit logs for forensic review.

- **Certificate and Key Management**
  The system shall provide tooling for secure provisioning, rotation, and revocation of certificates and keys, using protected keystores on edge devices to maintain fleet-wide trust
- **Differential Privacy (DP) Module**
  The client-side privacy module shall apply Differential Privacy with configurable clipping norms and tunable privacy budgets, exposing interfaces for adjusting and monitoring privacy–utility trade-offs

- **Homomorphic Encryption (HE) Module**
  The optional Homomorphic Encryption feature shall support encrypted model aggregation on the server, with adjustable parameters to balance security, computation cost, and communication overhead.
- **Configurable Privacy Modes**
  A unified policy layer shall allow per-task or per-dataset selection among FL, FL+DP, FL+HE, and FL+DP+HE configurations, ensuring consistent operation across clients and rounds.
- **Training and Model Validation Stack**
  The platform shall support standard deep learning frameworks and optimizers to perform local training, serialize updates, and validate model accuracy parity with non-secure baselines under IIoT conditions.
- **Messaging and Serialization**
  The platform shall use efficient, schema-validated serialization (e.g., JSON or Protobuf) compatible with the integrity and transport layers to ensure reliable and interoperable data exchange.
- **Configuration Management and Co-Tuning**
  The configuration system shall enable coordinated tuning of DP noise, HE parameters, aggregation thresholds, and detection sensitivity to meet desired trade-offs among privacy, accuracy, and latency.
- **Deployment and Operations**
  Automated workflows shall manage client enrollment/de-enrollment, certificate rotation, policy rollout, and incident mitigation, minimize downtime and ensure operational safety in production deployments.

These system software requirements establish a complete foundation for implementing and maintaining the integrated IIoT Federated Learning architecture. By combining secure communication, privacy protection, robust aggregation, and active defense measures, the platform ensures trustworthy and efficient learning under industrial restrictions. Each requirement will be validated through implementation and testing against quantifiable criteria accuracy, convergence, latency, bandwidth, resource utilization, and resilience to confirm that the framework satisfies both functional and operational goals in real-world IIoT implementations

3 METHODOLOGY

This chapter gives an overview of the implementation process of the components of the proposed solution architecture.

3.1 Overall System Architecture

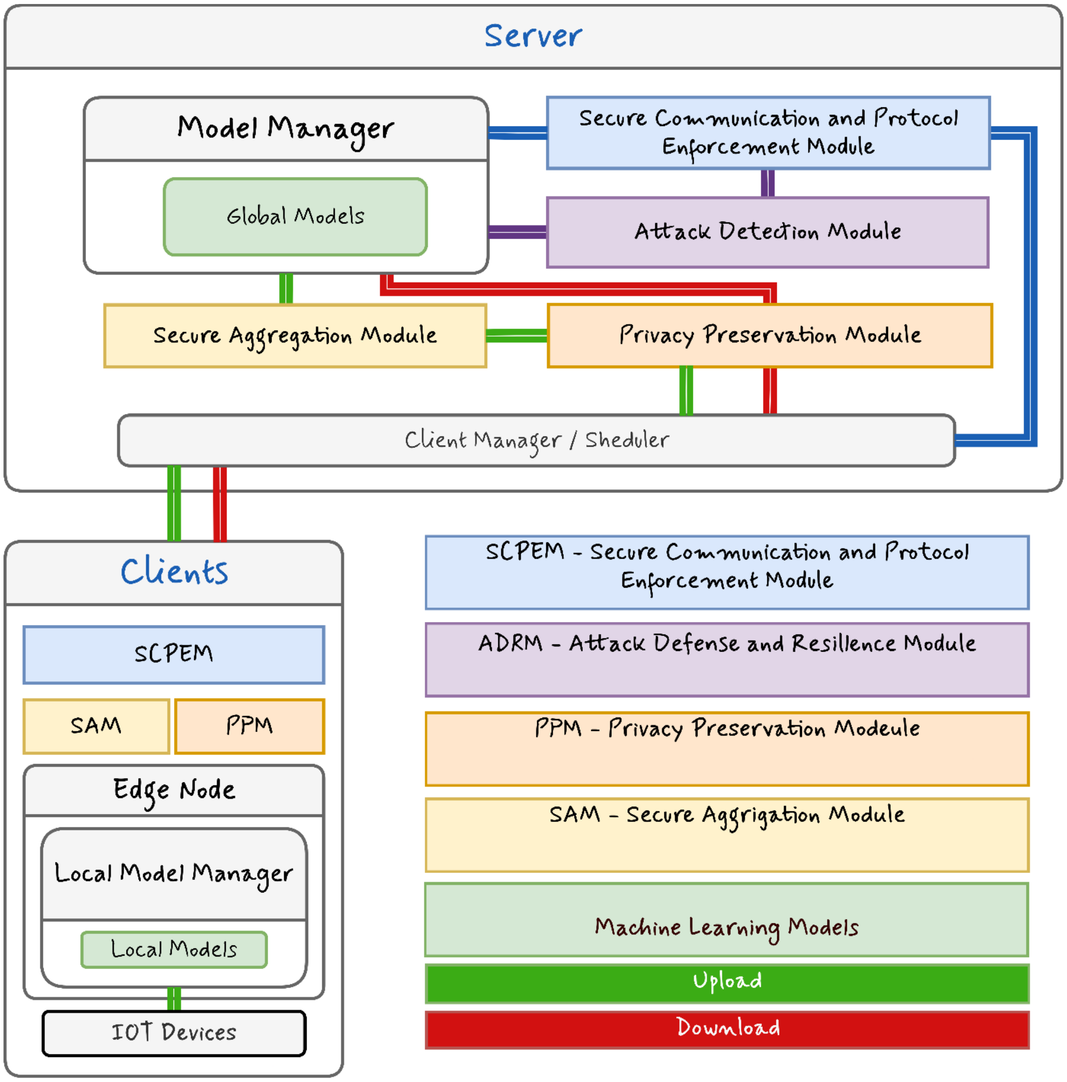

Server
Model Manager
Global Models
Secure Communication and Protocol Enforcement Module
Attack Detection Module
Secure Aggregation Module
Privacy Preservation Module
Client Manager / Sheduler
Clients
SCPEM
SAM
PPM
Edge Node
Local Model Manager
Local Models
IOT Devices
SCPEM - Secure Communication and Protocol Enforcement Module
ADRM - Attack Defense and Resillence Module
PPM - Privacy Preservation Modeule
SAM - Secure Aggrigation Module
Machine Learning Models
Upload
Download

Figure 1: System Architecture

As shown in, it was designed to manage the entire federated learning process, from client selection to model aggregation and management, while maintaining a strong focus on data privacy. The system is composed of several interconnected modules that work in synergy to achieve this objective.

The system is split into two main parts:

Server: A central coordinator that manages the overall learning process. It's responsible for aggregating model updates and distributing the new "global model." It does not receive the clients' raw data.

Clients: These are the edge devices (like IIoT devices or edge nodes) that hold the local data. They train models locally and send only the model updates (not the data itself) back to the server.

The entire system includes key security and privacy modules, which exist on both the server and client sides to ensure end-to-end protection:

- SCPEM (Secure Communication and Protocol Enforcement Module)
- ADRM (Attack Defense and Resilience Module)
- PPM (Privacy Preservation Module)
- SAM (Secure Aggregation Module)

Server Components

The Server orchestrates the federated learning process:

Client Manager / Scheduler: This component selects which clients will participate in a training round and coordinates the process.

Model Manager: This stores and manages the "Global Models." After aggregation, it updates this global model and distributes it to clients for the next round.

Secure Communication and Protocol Enforcement Module (SCPEM): This module acts as the gatekeeper for all communication. It ensures that any client connecting to the server is authenticated and that all data exchanged (like model uploads) is secure and follows the rules.

Privacy Preservation Module (PPM): This module works with the SAM to ensure the server can't "see" any individual client's model update, even in an encrypted or masked form.

Attack Detection Module (ADM): This module inspects the model updates it receives (after aggregation) to check for signs of malicious activity, such as data poisoning or backdoor attacks, protecting the global model from being corrupted.

Secure Aggregation Module (SAM): This is where the core federated learning "magic" happens. It receives protected model updates from many clients and combines (aggregates) them to create a single, improved model update. It is designed to work even if some clients drop out and ensures the server only learns the sum of the updates, not any individual one.

Client Components

The Client performs the local training on its own data:

IoT Devices: These are the sensors or machines that generate the raw data.

Edge Node: A component that manages the local learning process.

Local Model Manager: This holds the "Local Models." It receives the current global model from the server, trains it on the local data from the IoT devices, and then prepares the update (the change to the model, not the data) to be sent back.

SCPEM, SAM, and PPM: Just like on the server, these modules work together on the client side. Before sending the local model update, the PPM and SAM will encrypt or "mask" it to protect its privacy. The SCPEM then ensures this protected update is sent securely to the server.

Data Flow (How it Works)

The diagram shows the flow of information, color-coded by module:

Download (Red Line): The Client Manager on the server sends the current Global Model to the clients. This flow is protected by the SCPEM (blue line) and the SAM (red line).

Local Training (Inside Client): The client's Local Model Manager trains this global model on its local data, creating a new "local model update."

Upload (Green Line):

- The client's PPM and SAM (yellow/orange) first protect this local update.
- The update is then sent back to the server. This upload flow is protected by the SCPEM (blue line), processed by the PPM (orange line), and finally delivered to the Secure Aggregation Module (SAM) (green line).

Aggregation & Detection (Inside Server):

- The server's SAM gathers all the protected updates from clients and aggregates them into one.
- This aggregated update is passed to the Attack Detection Module (purple line) for inspection.
- If cleared, the update is sent to the Model Manager to improve the Global Model.

Repeat: The cycle repeats, with the new-and-improved global model being sent back to clients for the next round of training.

3.2 Secure Communication and Protocol Enforcement Module (SCPM)

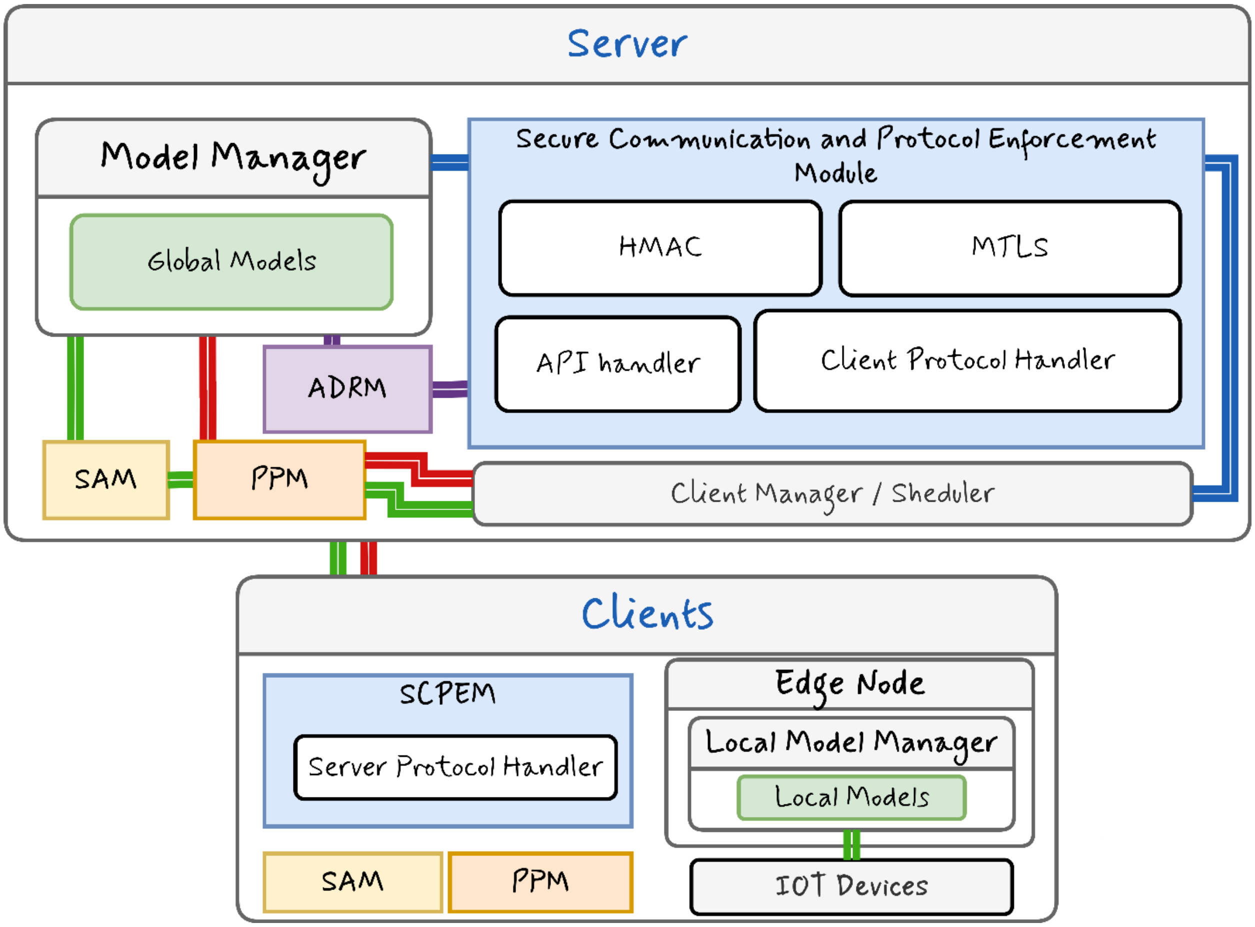


Figure 2: SCPM Module

This module is responsible for ensuring the integrity and confidentiality of data exchanged between the server and the clients. Both in the client and server this will be the rulebook, It will enforce secure communication protocols, such as TLS/SSL, to prevent eavesdropping and data tampering, and HMAC use to data-integrity while data transit. The SCPM also validates messages and ensures that all participants adhere to the defined communication protocols, preventing unauthorized access and malicious data injections.

- **System design**: delineate all federated learning control channels (model distribution, update submission, synchronization, and command and control directives), ascertain trust boundaries, and catalog potential threat actions (man-in-the-middle, replay, tampering, illegal command and control) for each message type.

- **Transport security:** implement TLS 1.3 with mutual authentication (client and server certificates) to encrypt all channels and facilitate Perfect Forward Secrecy, while checking certificate chains and revocation during session initiation.
- Ensure message integrity by appending HMAC to each request and response (including model blobs, control messages, and acknowledgments) and rejecting any mismatches to avert manipulation and replay attacks, incorporating nonce and timestamp verifications for freshness.
- **Protocol enforcement:** provide role-based access control to ensure that only authorized roles can enroll clients, commence aggregation, release models, or modify policies; enforce least privilege and document all policy decisions for auditing purposes.
- **Execution**: develop the security framework in Python utilizing standard TLS/HMAC libraries, integrate with the federated learning server and client stubs, and encapsulate validations into middleware to reduce modifications to the application.
- **Assessment**: replicate IIoT scenarios with limited clients and unreliable connections; quantify handshake latency, per-round overhead, CPU/memory effects, and performance.

3.3 Attack Detection and Resilience Module (ADRM)

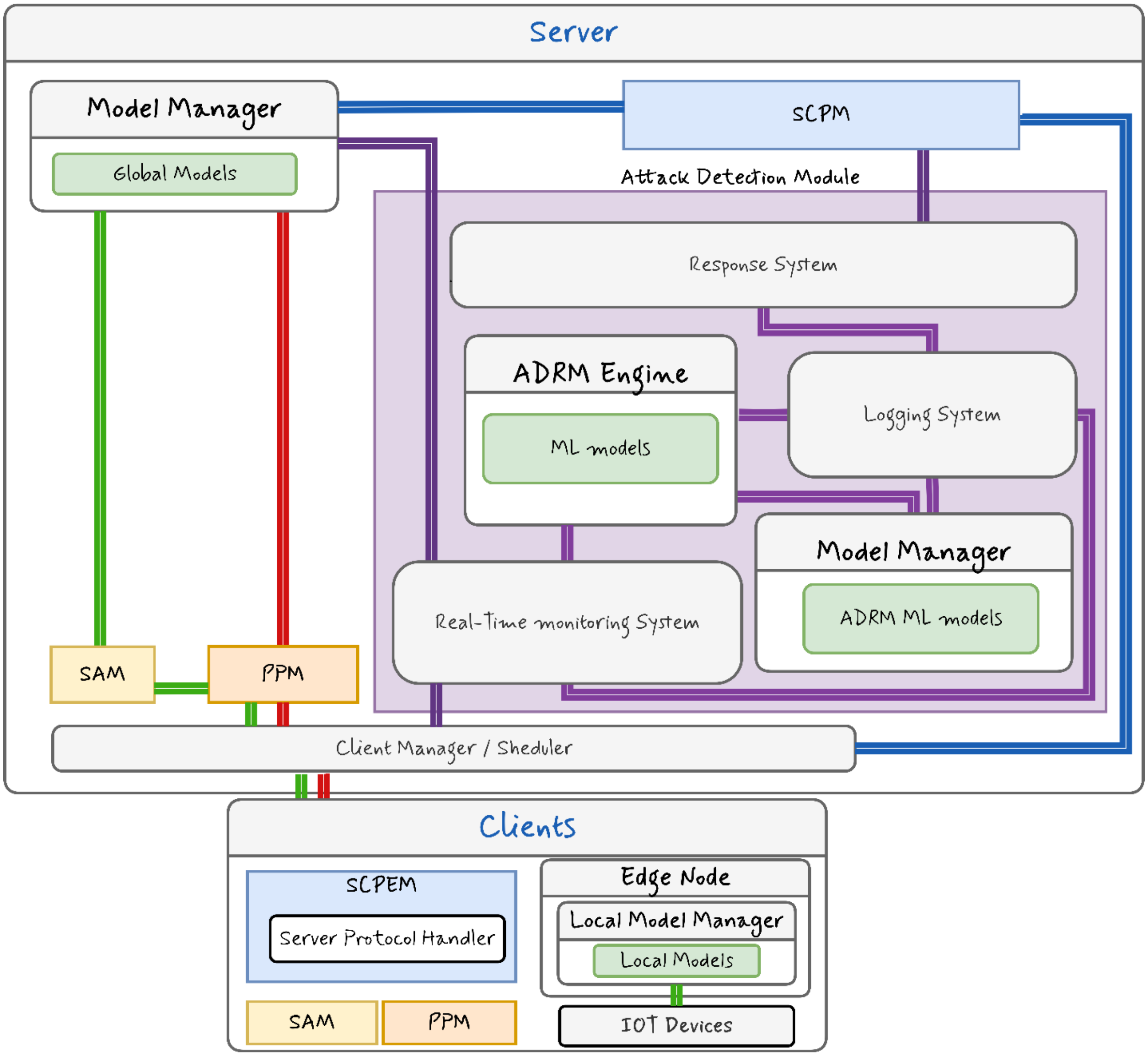


Figure 3: ADRM Module

This module is a critical security component that actively monitors the system for a wide range of attacks, including poisoning attacks, Byzantine attacks, free-riding, and inference attacks. It employs anomaly detection techniques and statistical analysis to identify suspicious client behavior or corrupted model updates. When an attack is detected, the ADRM takes measures to mitigate its impact, such as isolating malicious clients, reducing trust scores or discarding compromised updates, thereby enhancing the system's resilience.

- **Architecture**: place a lightweight anomaly detection pipeline on the server side of the FL loop to score incoming updates before aggregation, with actions to down-weight, reject, or quarantine suspicious clients.

- **Feature design:** extract update-level features (e.g., gradient norms, cosine similarity to median, update direction outliers) and round-level signals (e.g., variance spikes) to capture poisoning and Byzantine behaviors.
- **Detection models:** use efficient detectors suitable for real time on constrained infrastructure (e.g., rules plus lightweight ML filters), tune thresholds for non-IID data, and maintain whitelists/blacklists with decay policies.
- **Response and recovery:** integrate robust aggregation, automated rollback of the global model when needed, and adaptive retraining to restore model integrity after detected incidents.
- **Evaluation**: simulate attacks (label-flipping, model/backdoor poisoning, Byzantine updates) and measure detection precision/recall, time-to-mitigation, sustained accuracy, and latency/overhead across rounds and client scales.

3.4 Privacy Preservation Module (PPM)

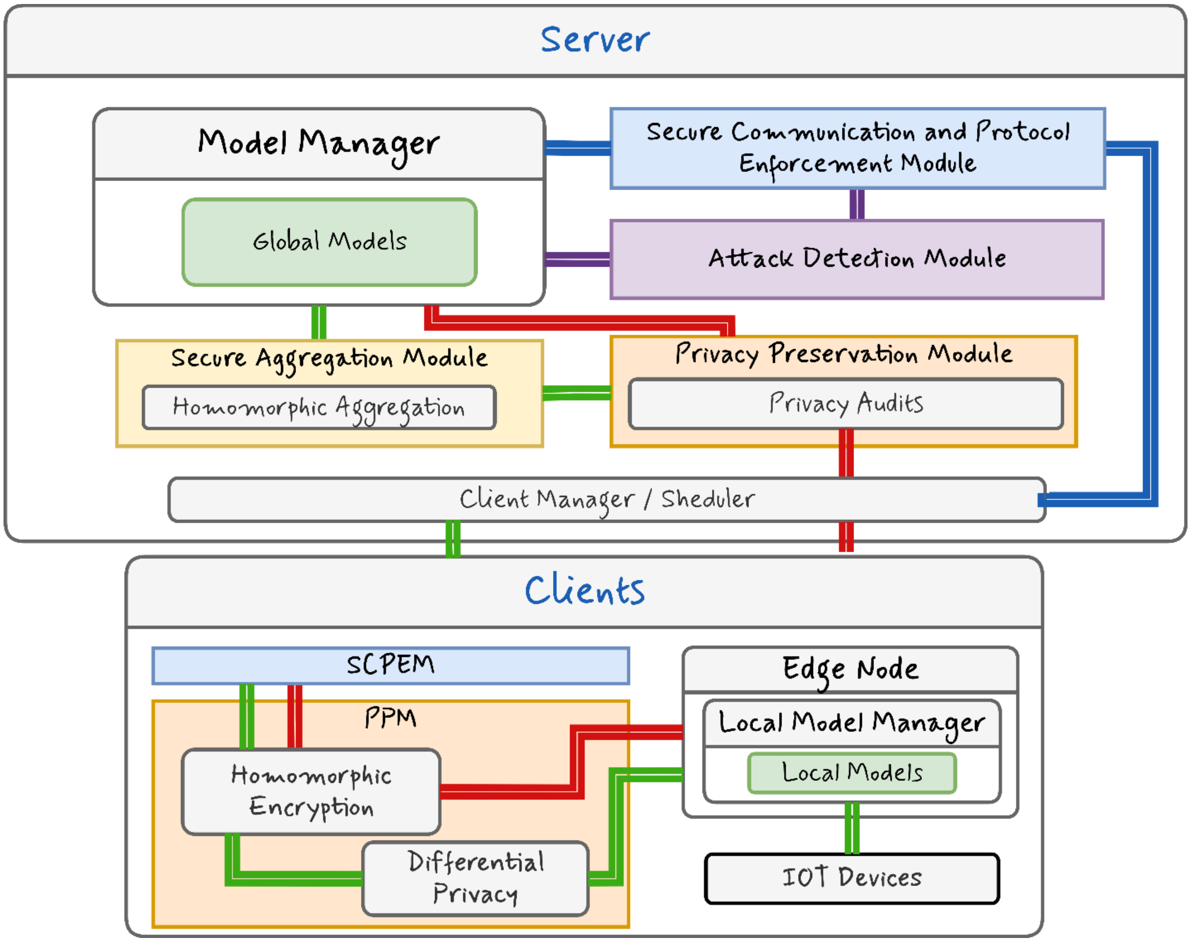


Figure 4: PPM Module

This module is at the core of the system's theta-privacy focus. It implements various privacy-enhancing techniques, such as differential privacy (DP), homomorphic encryption (HE), and secure multiparty computation (SMC). The module prevents the central server or other clients from accessing sensitive information by adding noise to model updates or by computing on encrypted data. It also includes privacy audits to verify that the privacy mechanisms are functioning correctly

- **Baseline FL**: put up a typical FL pipeline with non-IID client data, fixed rounds/epochs, and FedAvg aggregation to establish accuracy, convergence, and communication baselines for IIoT workloads.
- **Differential Privacy:** implement per-client gradient clipping and Gaussian noise addition with customizable privacy budgets, and record accuracy vs. privacy trade-offs and message size changes owing to DP metadata.

- **Homomorphic Encryption:** encrypt client model changes, execute server-side encrypted aggregation, then decrypt the aggregate for the global update; track ciphertext expansion, compute overhead, and end-to-end delay.
- **Hybrid evaluation:** compare FL, FL+DP, FL+HE, and FL+DP+HE on industrial datasets for accuracy, bandwidth, latency, and device resource utilization; provide guidelines for budget selection and encryption parameters under IIoT limitations.
- **Deployment guidance:** document appropriate ε ranges, clipping standards, HE schemes/parameters, and when to prefer DP, HE, or a hybrid based on privacy requirements, device capability, and network restrictions.

3.5 Secure Aggregation Module (SAM)

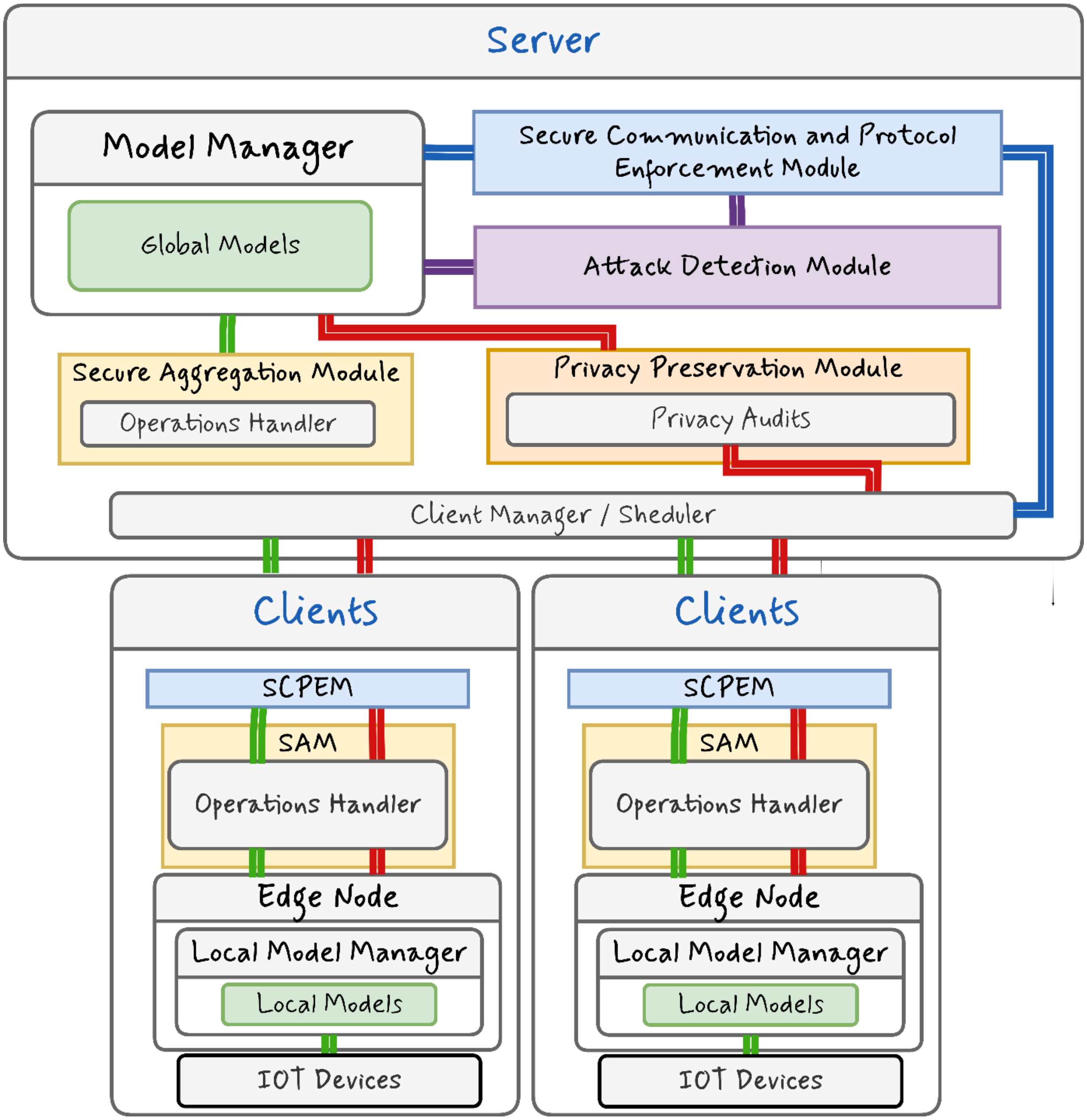


Figure 5: SAM Module

The module works with the Privacy Preservation Module (PPM) to ensure the central server cannot inspect raw updates. The SAM utilizes a dual-protocol approach to achieve this: Shamir's Secret Sharing, which divides updates into reconstructible shares to prevent single points of failure, and the Bonawitz Protocol, a robust, federated learning-specific method that enables the server to compute the sum of encrypted updates without decrypting them. This combination guarantees that the aggregated model is a true and private sum of all valid contributions, even from the central server itself.

- **Protocol design:** implement a two-layer secure aggregation protocol using pairwise masking to hide individual updates and Shamir's Secret Sharing (SSS) to reconstruct and remove masks for dropped clients, ensuring only the sum is revealed.
- **Resource-aware masking:** introduce adjustable mask precision (e.g., 32- vs 64-bit) to match device capabilities while preserving privacy and correctness and specify compatibility rules for mixed-precision participation.
- **System integration:** realize client workflows for key exchange, share distribution, masked update submission, and threshold recovery; integrate with the server's aggregation loop and error handling for dropouts.
- **Tools and stack:** use Python with gRPC/Protobuf for multi-round messaging, PyTorch for model training, and cryptographic libraries for pairwise keys and SSS primitives to support iterative FL rounds.
- **Evaluation:** test threshold recovery across varying dropout rates, verify final model parity with non-secure baselines, and assess resilience under backdoor/poisoning attempts alongside bandwidth and compute overhead.

3.6 Manager Components

- **Client Manager**
  Handler of the client participation, including selection and scheduling for training rounds. based on criteria like availability and network status. It ensures timely participation and manages client enrollment and state

- **Orchestration**
  As the central coordinator, the module manages the entire federated learning process from start to finish. It coordinates with all other components to ensure a seamless workflow and enforces security and privacy protocols throughout the training life cycle.

- **Model Manager**
  Created and dedicated to managing the shared machine learning model. It stores versions and distributes the global model to clients. It also receives the aggregated updates from the Secure Aggregation Module (SAM) and applies them, ensuring the integrity and consistency of the model.
- **Log Manager**

Provider of a comprehensive record of system activities. It collects timestamps, and stores logs from all other components, including communication events and detected anomalies. These logs are essential for monitoring, security audits, and debugging the system.

3.7 Commercialization

The project's commercialization approach focuses on building an open-source federated learning platform purpose-built for Industrial IoT (IIoT) contexts. The platform is designed to maximize confidence, openness, and adoption in areas where data protection, operational integrity, and compliance are crucial. By providing the whole stack including secure communication protocols, privacy-preserving aggregation methods, and built-in anomaly defense as open source, the system enables direct verification, peer audit, and rapid adaptation to real-world security needs. This method differentiates it from closed, opaque systems while satisfying the unique technical and regulatory constraints of industrial processes.

Target Audience

- Operators of Industrial IoT (IIoT), producers, and vendors of plant automation.
- Research institutions and security laboratories specializing in privacy-preserving machine learning.
- Government agencies need auditable and self-hosted FL infrastructure.
- System integrators and managed service providers within the domains of data security and analytics.

Value Proposition

- Transparency & Trust: Open-source design ensures verifiability and security confidence.
- Customization & Control: Operators can self-host, alter, and integrate into existing IT/OT systems.
- Industrial Readiness: Pre-integrated RBAC, message authentication, and privacy (DP/HE) components effectively tackle practical IIoT limitations.
- Scalable Security: Secure aggregation and adaptive defense mechanisms change with threat landscapes.

Revenue & Sustainability Model

- Support & Integration Services: Paid deployment, customization, and technical training for industrial clients.

- Public & Research Grants: Funding through programs supporting digital security, industrial innovation, and privacy-preserving analytics.
- Corporate & Institutional Sponsorships: Joint development and funding with industry partners and government bodies.

Partnership & Growth Plan

- Collaborate with industrial automation corporations, IIoT consortiums, and research bodies to increase adoption and interoperability.
- Establish pilot installations in genuine industrial situations to demonstrate scalability and reliability.
- Foster a community-driven ecosystem for ongoing improvement and vulnerability reporting.
- Engage with regulatory and standards bodies to match the platform with emerging compliance and data-protection regimes.

3.8 Testing And Implementation

technical implementation of the proposed Data-Privacy Focused Federated Learning Framework for Industrial IoT (IIoT) and the empirical evaluation of its performance. The framework was developed as a cohesive client-server system, integrating four specialized modules to provide a multi-layered defense-in-depth security model. The implementation was conducted in a simulated IIoT environment, with a server orchestrating a pool of heterogeneous clients. The core system was developed in Python (v3.10+), leveraging powerful, industry-standard libraries to achieve its objectives.

3.8.1 Development Environment and Tools

The framework's implementation relied on the following key technologies:

- **Machine Learning**: PyTorch was the primary library for defining, training, and testing the Convolutional Neural Network (CNN) models. NumPy and Pandas were used for high-performance numerical operations and data manipulation.
- **Communication:** gRPC with Protocol Buffers was used to establish an efficient, low-latency, and asynchronous communication channel between the server and clients.
- Security & Privacy:
  - **Opacus**: A PyTorch-affiliated library used to implement Differential Privacy (DP), automating gradient clipping and noise injection.
  - **Paillier**: A Python library for the Paillier cryptosystem was used to implement additively Homomorphic Encryption (HE).

    - **pyOpenSSL & cryptography**: These libraries were used to implement TLS 1.3 and HMAC for the secure communication module.
    - **secret-sharing & PyCryptodome**: These were used to implement Shamir's Secret Sharing (SSS) and other cryptographic primitives for the secure aggregation module.
- **Simulation:** The IIoT environment, comprised of multiple client nodes, was simulated using tools including VirtualBox and Docker.
- **Monitoring:** A custom Terminal User Interface (TUI) was developed using the Rich and *AIOHTTP* libraries to provide real-time, at-a-glance monitoring of the server status, active clients, and module performance

3.8.2 System Architecture and Module Implementation

The framework is built on a client-server architecture, with the server hosting the global model and coordinating clients, which represent IIoT edge nodes. The implementation integrates the four specialized modules at different layers of this architecture.

- **Secure Communication and Protocol Enforcement Module**

This module secures the fundamental communication channel (the gRPC-based C2 channel) between the server and all clients. Its implementation ensures:

- **Confidentiality & Authentication:** The channel is secured using TLS 1.3 with mutual authentication (mTLS) . Both the server and clients must present valid, CA-signed certificates to establish a connection, preventing unauthorized devices and Man-in-the-Middle (MitM) attacks.
- **Data Integrity:** Hash-based Message Authentication Code (HMAC) is implemented to validate the integrity and authenticity of every message. The receiver rejects any message with an invalid HMAC tag, protecting against data tampering.
- **Access Control:** A Role-Based Access Control (RBAC) mechanism is enforced at the protocol level. This ensures that an authenticated device can only execute commands permitted for its role (e.g., a "client" role can send updates but not initiate an aggregation round).

- **Privacy Preservation Module**

This module protects the content of the model updates from being exposed to the server.

- **Differential Privacy (DP):** Implemented using the Opacus library, DP adds calibrated Gaussian noise to model updates before they leave the client device. This provides a formal, mathematical privacy guarantee. The implementation also automated gradient clipping (clipping norm set to 1.0) to limit the influence of any single data point.

- **Homomorphic Encryption (HE):** The Paillier cryptosystem was implemented to provide an additional layer of confidentiality. Clients encrypt their DP-protected updates with a public key. The server, which does not have the private key, performs aggregation by summing the encrypted values. Only a trusted authority with the private key can decrypt the final aggregated result .

- **Secure Aggregation Module**

This module ensures the federated learning process is resilient to client failures, which are common in unstable IIoT networks.

- **Privacy (Masking):** The module implements pairwise masking, where clients establish shared secrets to create cryptographic masks. One client in a pair adds the mask, and the other subtracts it. This ensures all masks mathematically cancel out upon aggregation, hiding individual contributions from the server.
- **Robustness (SSS):** To handle client dropouts (which would leave masks uncancelled), the module implements a threshold-based recovery mechanism using Shamir's Secret Sharing (SSS). Clients distribute "shares" of their secrets. If a client drops out, the server can collect a threshold of shares from the remaining online clients to reconstruct and remove the missing client's mask, allowing the aggregation to complete successfully.

- **Attack Defense and Resilience Module (ADRM)**

The ADRM is the framework's active defense, designed to protect the global model's integrity from malicious clients attempting poisoning or Byzantine attacks.

Anomaly Detection: The module was implemented as a real-time detection engine that intercepts client updates. It uses unsupervised learning to identify malicious updates, such as those with large gradient deviations or inconsistencies.

Lightweight Model: A lightweight Isolation Forest model was implemented for this task. This model was chosen for its high accuracy in detecting outliers and its low computational overhead, making it ideal for a real-time server-side scanner

3.8.3 Testing and Evaluation

The integrated framework was tested in a simulated environment using two primary datasets: CIFAR-10 for general accuracy and privacy-tradeoff analysis, and a custom Industrial IoT dataset for anomaly detection performance.

- **Security and Privacy (SCPEM & PPM)**

The privacy-accuracy trade-off was evaluated using the CIFAR-10 dataset over 10 training rounds.

- **Baseline (Standard FL):** The model achieved a baseline accuracy of approximately **76%.**

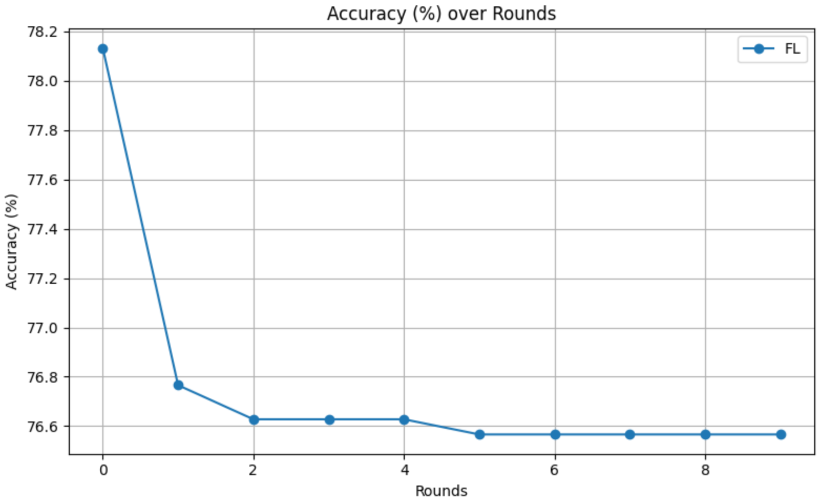


Figure 6: FL Accuracy over Rounds

- **FL + DP:** With Differential Privacy enabled, the introduced noise reduced the final accuracy to ~65%, demonstrating the expected trade-off.

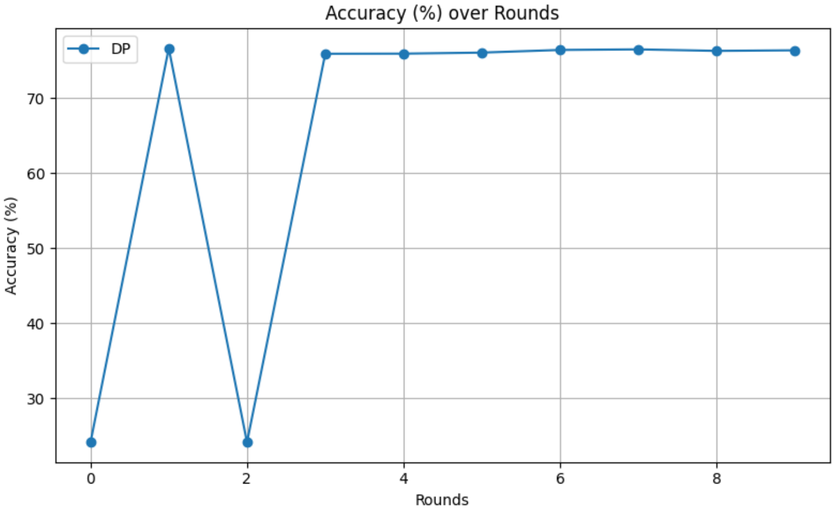


Figure 7: FL + DP Accuracy over Rounds

- **FL + HE:** Homomorphic Encryption successfully maintained high accuracy **(~76%)** , but this came at the cost of a 5-10x increase in communication overhead due to ciphertext expansion.

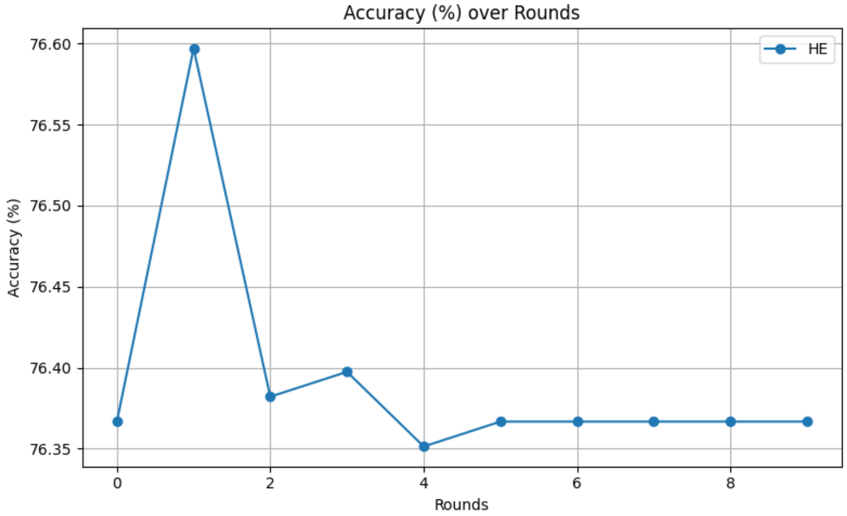


Figure 8: FL + HE Accuracy over Rounds

- **FL + DP + HE:** The maximum privacy configuration (combining both techniques) yielded an accuracy of **~75%.**

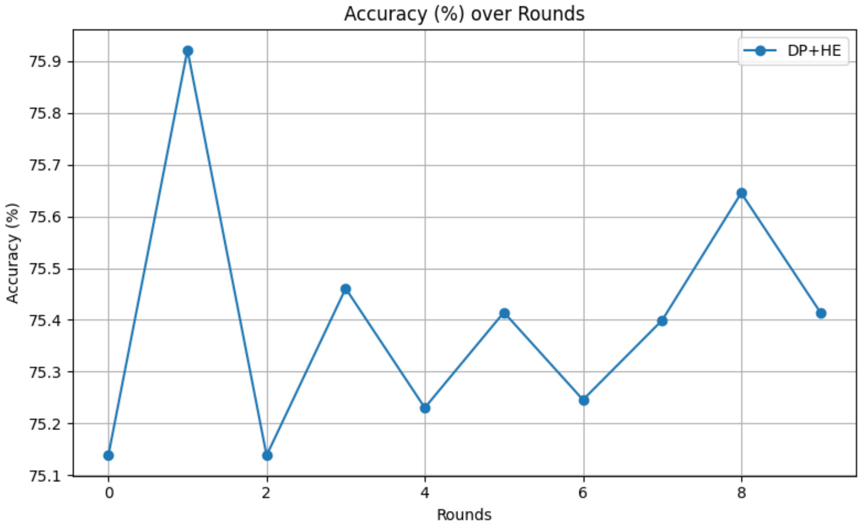


Figure 9: FL + DP + HE Accuracy over Rounds

- **Communication Security:** The SCPEM module was validated to successfully block unauthorized (unauthenticated) connection attempts and reject messages that were tampered with (failing the HMAC check).

- **Security and Privacy (SCPEM & PPM)**

The framework's resilience to network failures and malicious attacks was tested.

Fault Tolerance (SAM): The Secure Aggregation Module's threshold recovery was highly effective. In simulations, the framework achieved a 100% aggregation success rate even when 30% of the clients failed to respond in a given round.

Attack Defense (ADRM): The ADRM's performance was tested in an adversarial scenario with 10% malicious clients.

Without ADRM: Under a model poisoning attack, the baseline model's accuracy quickly collapsed to less than 20%. The backdoor attack success rate peaked at over 40%.

With ADRM: The ADRM, combined with a robust aggregation algorithm (Krum), successfully mitigated the attack. It maintained a high global model accuracy of over 85% and kept the backdoor attack success rate near 0% in the initial rounds, never allowing it to exceed 28%.

Performance: The ADRM proved to be lightweight, adding only minimal latency (e.g., 40ms per round vs. 35ms baseline) and negligible CPU overhead

4 RESULTS AND DISCUSSION

4.1 Results

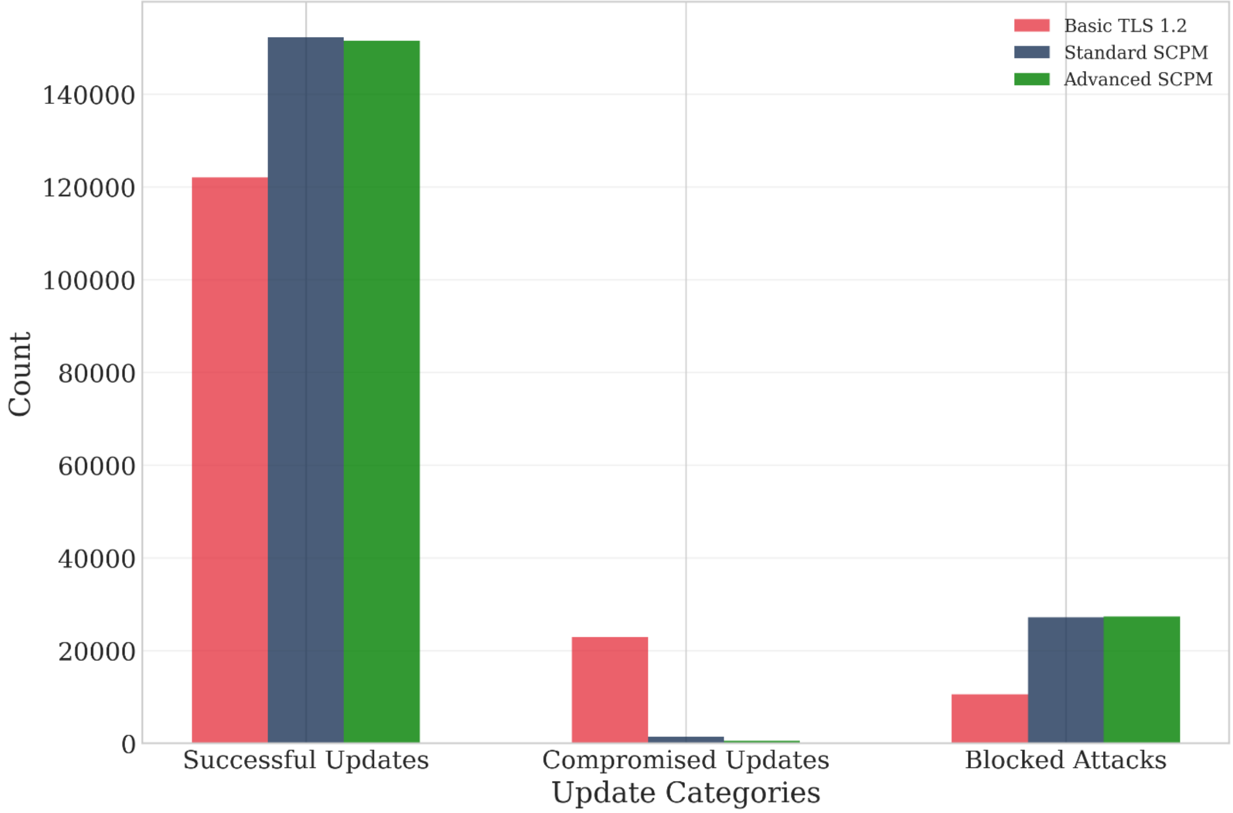


Figure 10: Effectiveness of different strategies

The chart illustrates the effectiveness of different secure communication and protocol enforcement strategies against malicious activity during federated learning in IIoT settings. Three approaches are evaluated: Basic TLS 1.2, Standard SCPM (Secure Communication and Protocol Module), and Advanced SCPM.

On the x-axis, three categories are represented: Successful Updates, Compromised Updates, and Blocked Attacks. The y-axis shows the count for each category across multiple training rounds.

Results clearly demonstrate that both SCPM variants (Standard and Advanced) outperform Basic TLS 1.2 in terms of both model integrity and resilience to attacks:

- **Successful Updates:** Advanced SCPM and Standard SCPM maintain the highest number of successful model updates, reaching totals well above Basic TLS 1.2. This indicates that robust protocol enforcement and mutual authentication can facilitate resilient, uninterrupted federated learning even under adversarial conditions.

- **Compromised Updates:** The number of compromised updates—those that were successfully tampered with or poisoned—is dramatically reduced in both SCPM strategies compared to Basic TLS 1.2. The bars for Standard and Advanced SCPM are almost negligible, showing their effectiveness at detecting and deflecting malicious traffic.
- **Blocked Attacks:** Advanced SCPM and Standard SCPM successfully block a much greater number of attacks than Basic TLS 1.2. The increased blocked attack count reflects proactive monitoring, strong access control, and protocol enforcement introduced in SCPM, making it far more difficult for adversaries to inject or manipulate updates into the global model.

the results clearly suggest the implementation of a specialized Secure Communication and Protocol Module especially with enhanced mutual authentication, HMAC validation, and granular RBAC controls for any industrial federated learning deployment. These approaches give improved system robustness, drastically minimize successful breaches, and enable the active blocking of adversarial influence, which combined assist preserve data privacy, model integrity, and overall operational resilience for IIoT.

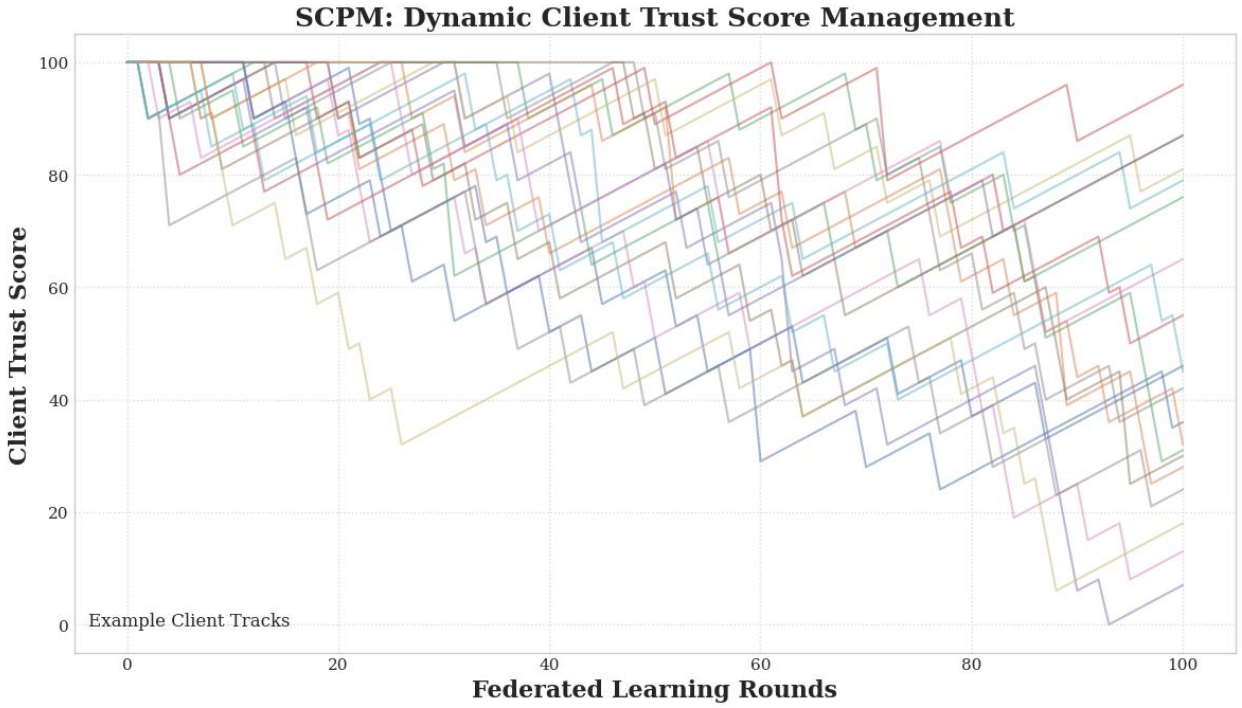


Figure 11: Client Trust Score over FL Rounds

This experiment demonstrates the effectiveness of the SCPM in enforcing protocol rules and upholding security in real time by dynamically scoring clients. The system's capacity to modify trust levels ensures that only dependable devices continue to significantly influence the federated model, while untrustworthy or compromised nodes are more separated and their impact on the aggregate is diminished. This approach is vital for maintaining model integrity and dependability in industrial dispersed learning contexts, where the danger of device compromise or insider threats is significant and rapid response is necessary.

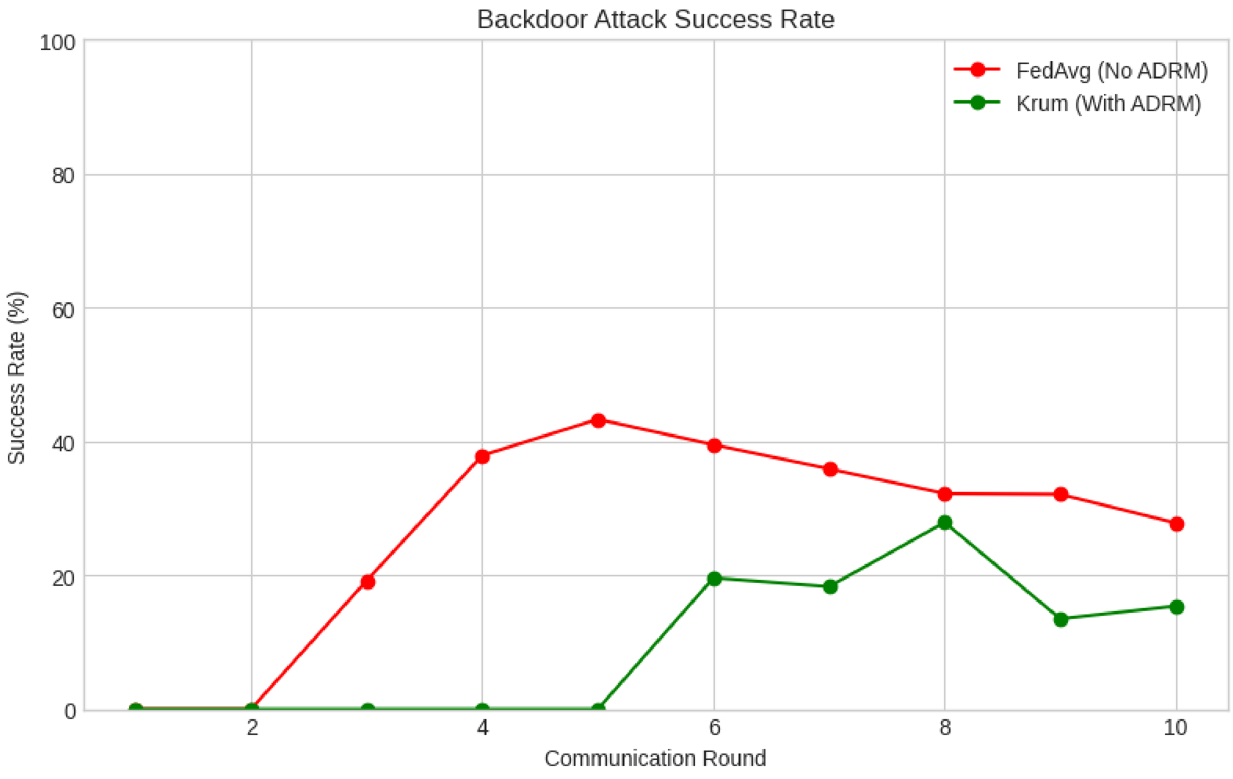


Figure 12: Communication Rounds over Success Rate

The "Backdoor Attack Success Rate" graph provides a direct comparison between two federated learning aggregation strategies under adversarial conditions across ten communication rounds: the standard Federated Averaging (FedAvg) aggregator with no attack detection/resilience module (ADRM), and the Krum aggregator enhanced with ADRM.

During the experiment, a gang of hostile customers attempts to poison the learning process by putting backdoors into the global model. The red line, showing typical FedAvg, shows a rapid increase in backdoor attack success—climbing to over 40% within a few rounds and continuing over 30% throughout. This finding supports the fragility of ordinary FL protocols; without effective defense, hostile clients can reliably breach the system and keep control of the model's behavior for several rounds.

In stark contrast, the green line (Krum + ADRM) indicates how the integration of strong aggregation and real-time attack detection drastically reduces attack success rates. For the majority of rounds, the backdoor success rate is held close to zero, and only modest increases occur in later rounds—these never approach the peaks seen in the FedAvg case. This protective effect arises from ADRM's mix of statistical analysis and anomaly detection, which allow the system to instantly detect and filter harmful or questionable inputs before they may alter the global model. The Krum aggregator further boosts resistance by down-weighting or rejecting outlier updates even when faced with sophisticated poisoning techniques.

In summary, our experiment conclusively illustrates that robust aggregation paired with dynamic anomaly detection is crucial for federated learning in real-world IIoT, as it tackles a spectrum of hostile threats with high effectiveness and operational practicality. The considerable disparity between the FedAvg and Krum (with ADRM) lines on this graph presents visual and statistical proof that the suggested group framework makes substantial advancements in model integrity and security for industrial machine learning applications.

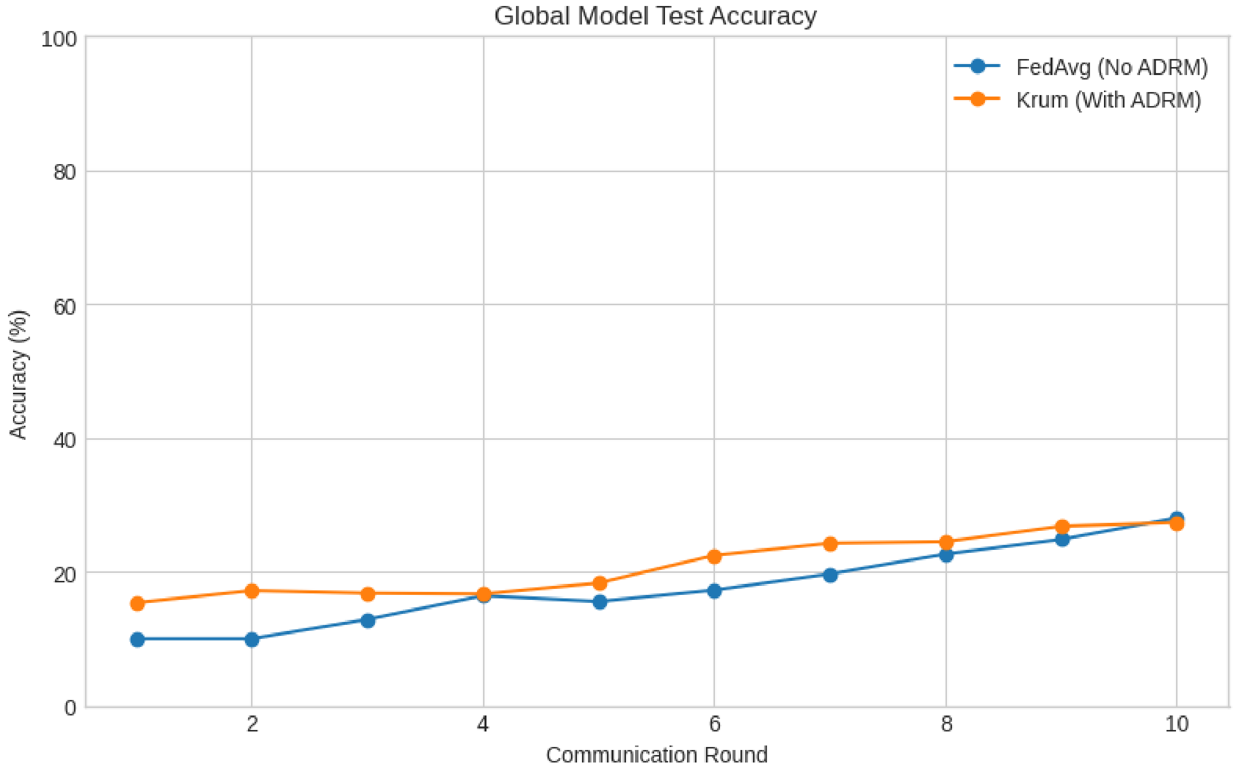


Figure 13: Communication Rounds over Accuracy

The "Global Model Test Accuracy" illustration offers a comparison of two aggregation schemes—standard FedAvg (No ADRM) and Krum (With ADRM)—over ten federated learning rounds in the presence of adverse clients. The accuracy (%), assessed on a held-out test set, is recorded on the y-axis, while the x-axis lists each FL communication round.

The results reveal that the system integrating the Attack Detection and Resilience Module (ADRM) with robust aggregation (Krum) regularly delivers higher global model test accuracy compared to the baseline FedAvg approach. In the opening rounds, Krum displays a stronger and more constant development, keeping or exceeding FedAvg at every point. As more rounds progress, the accuracy margin either remains consistent or grows significantly, with Krum attaining about 30% accuracy by round 10, while FedAvg trails slightly behind.

This performance boost is directly related to the ADRM's success at filtering out poisoned, Byzantine, or anomalous updates by recognizing and down-weighting hostile contributions before global aggregate. The baseline (FedAvg) is far more susceptible to the detrimental influence of malicious clients. As a result, the aggregate global model's quality and test performance decline, but Krum's robustness leads to constant accuracy improvements, even as adversarial pressure persists.

In the thesis's discussion, it's underlined that this margin is not only a statistical artifact. The ADRM approach, by integrating statistical anomaly detection, trust scoring, and resilient aggregation, consistently preserves model integrity and supports stable accuracy improvements equivalent to or perhaps exceeding those available in simply benign or controlled situations. Furthermore, the thesis emphasizes that these gains are accomplished without large increases in compute or communication overhead, supporting the suggested approach's applicability for resource-constrained IIoT applications.

In summary, this experimental result visually and empirically validates the assumption that robust aggregation and real-time anomaly detection are critical for maintaining model performance in federated learning deployments faced with adversarial hazards. The continuous advantage of Krum (with ADRM) across rounds offers significant support for using these techniques in future industrial ML systems.

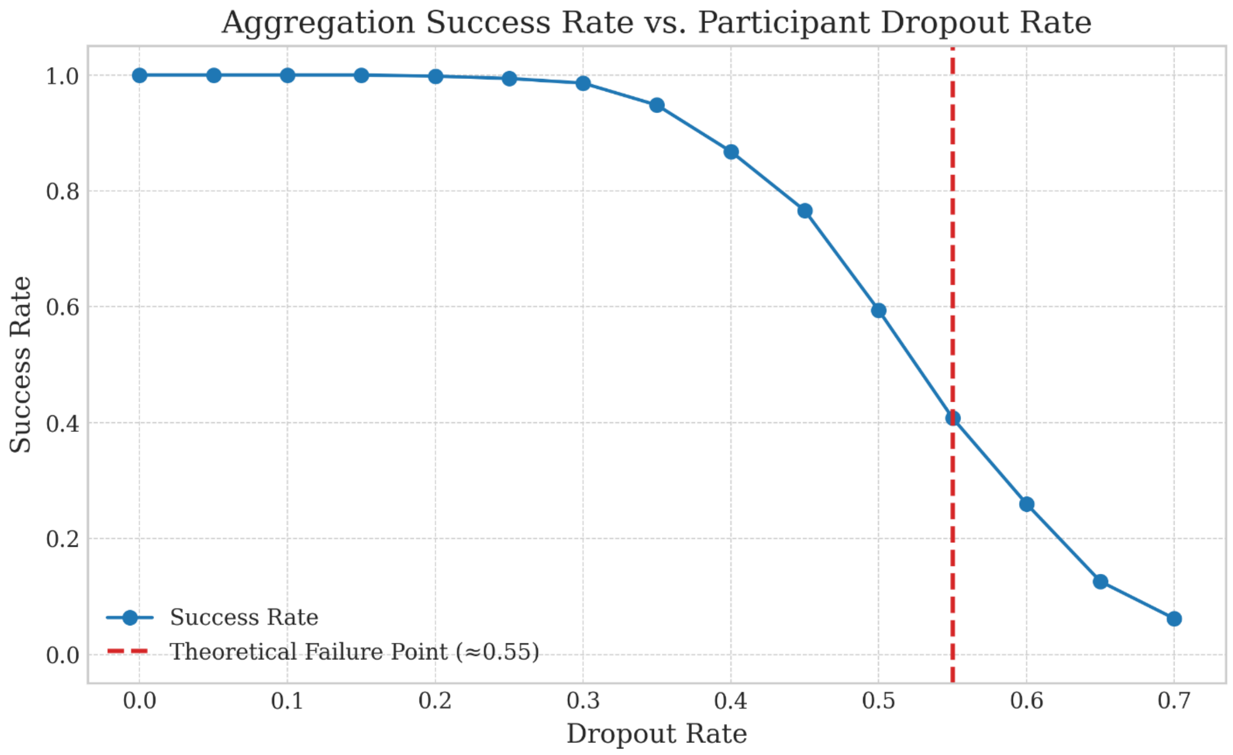


Figure 14 : Aggregation Success Rate vs Participant Dropout Rate

The graph shows how the aggregation success rate drops as the participant dropout rate increases in the federated learning process. Up to about 40% dropout, the system's success rate stays very high, meaning that aggregation can be reliably completed even if a significant portion of clients are lost during a training round. However, after the dropout rate passes roughly 0.55, marked by the red dashed line as the theoretical failure point for the threshold cryptography, the success rate falls rapidly. This matches the expected behavior of a secret sharing-based aggregation protocol, where a minimum number of shares is required to successfully reconstruct the sum of updates. The region before the threshold demonstrates strong fault tolerance and reliability in real-world Industrial IoT scenarios with network disruption or device failure. Beyond this dropout rate, the protocol cannot guarantee global model updates, as not enough client shares are available for correct aggregation. The experimental result directly confirms the protocol's mathematical properties and sets a clear operational limit for secure aggregation under client dropout in resource-constrained or unstable environments.

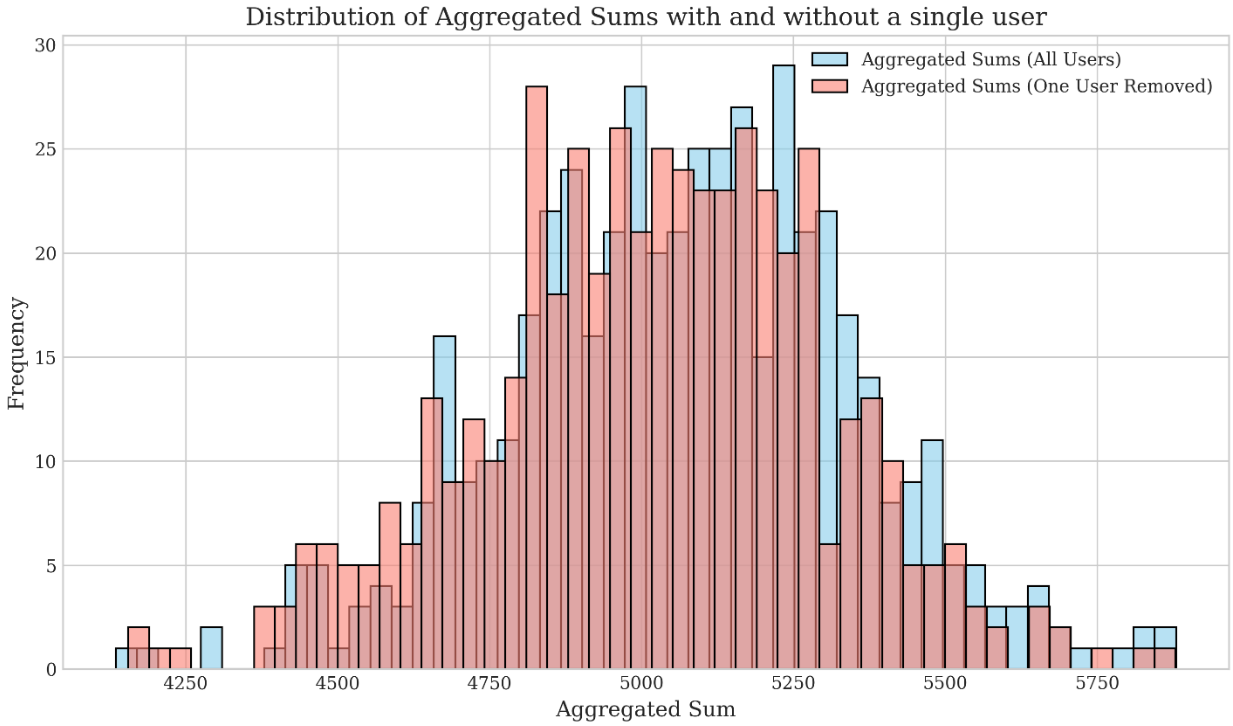


Figure 15 : Distribution of Aggregated Sum

The histogram examines the distribution of aggregated sums when all users engage in the secure aggregation protocol vs when one user is removed. The distributions stay essentially equal regardless of whether all users are present or a single user is missing, suggesting that the absence of any participant has a negligible effect on the aggregated outcome. This strong overlap demonstrates the protocol's central privacy property: individual contributions are mathematically masked out within the group, so even if an attacker or server observes the output before and after someone drops out, they cannot infer any unique information about that user's model update.

The experiment was aimed at verifying the protocol's resistance to privacy leakage by simulating aggregation under two scenarios and plotting the resulting sums. The outcome proves that the system secures user data by preventing isolated statistics from being learned via aggregated outputs if any single user's data made a noticeable difference in the sum, the two histograms would shift apart, enabling privacy breaches. Instead, the indistinguishability presented here indicates that the protocol both respects privacy and maintains the value of the federated model: model correctness is not destroyed, and dropout indistinguishability is obtained. This secures model updates in practice, as neither an external attacker nor the server can leverage aggregation changes across rounds to reconstruct private client information, even if numerous rounds or targeted dropouts are observed. In summary, the protocol provides privacy-respecting collaborative learning with robust aggregation unaffected by usual client loss in IIoT settings.

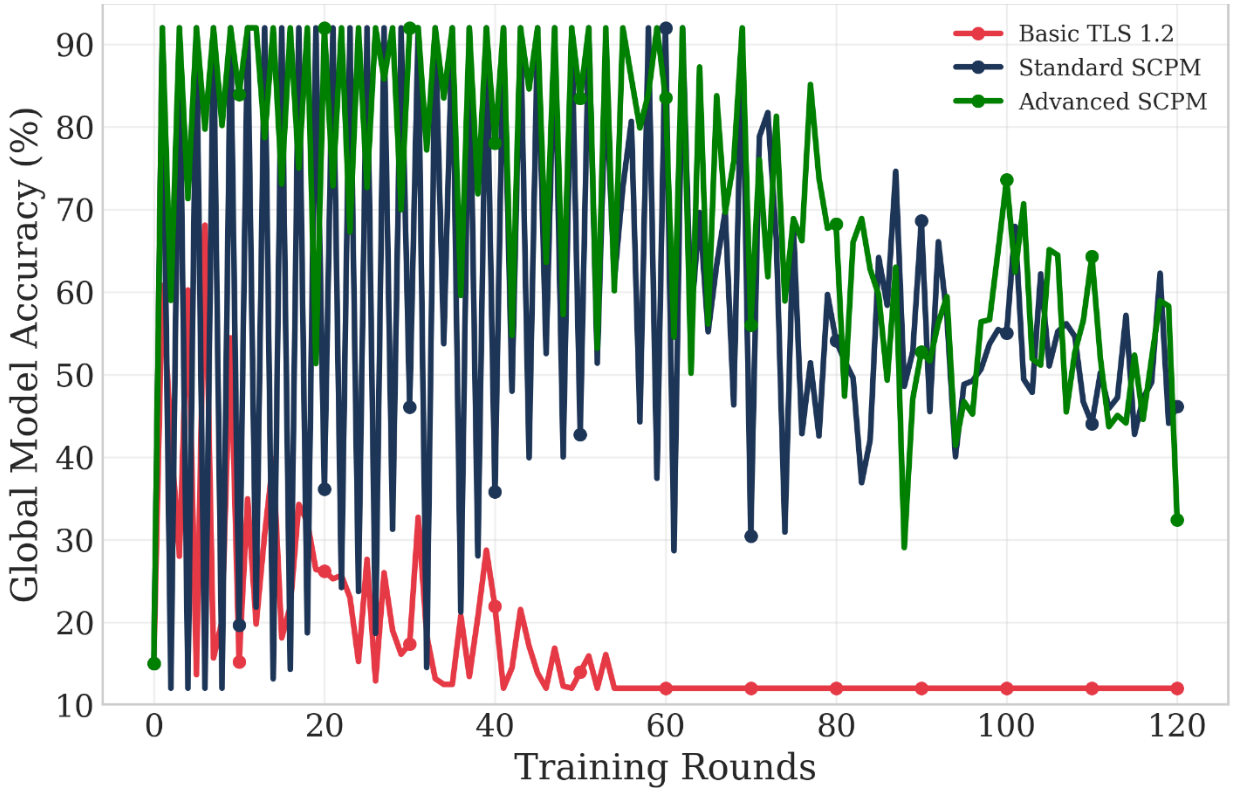


Figure 16 : Global Model Accuracy

The graph shows global model accuracy over 120 training rounds for three setups: Basic TLS 1.2, Standard SCPM, and Advanced SCPM. The results highlight substantial differences in FL model robustness depending on the security and protocol enforcement mechanisms. With only Basic TLS 1.2, model accuracy plunges quickly after the first few rounds, indicating the global model is quickly corrupted under adversarial pressure and remains unusable, consistently failing to surpass 20% accuracy. In contrast, when using Standard SCPM which builds on secure communication, protocol enforcement, and message integrity accuracy remains higher but is still volatile and occasionally subject to sharp drops, reflecting susceptibility to advanced attacks or network perturbations not fully mitigated by standard measures.

Advanced SCPM, which adds mutual authentication, HMAC integrity checks, and fine-grained access control, produces the most stable and highest accuracy across the experiment. The global model consistently stays above 70%, frequently exceeding 80–90%, and is dramatically less volatile compared to both Basic TLS 1.2 and Standard SCPM. This emphasizes the value of an integrated protocol stack in IIoT federated learning: securing every aspect of client-server communication, enforcing strict authentication and role-based access, and validating update integrity all work together to block injection, replay, and data tampering attacks that undermine model quality.

The overall conclusion is that models trained with robust, advanced SCPM maintain operational utility in hostile or unreliable environments, enabling secure, privacy-preserving machine learning for industrial systems without sacrificing performance or system resilience. These results empirically support the claim that multi-layered security enforcement is necessary for trustworthy federated learning at scale in IIoT deployments.

4.2 Research findings & Discussion

The research findings reveal that the proposed modular federated learning system enables robust, privacy-preserving, and secure machine learning for Industrial IoT even under adversarial and resource-constrained settings. Experimental results show that the Secure Communication and Protocol Enforcement Module (SCPM) provide considerable gains in system resilience, reducing the frequency of compromised updates and rejecting many more assaults compared to standard TLS 1.2. With advanced SCPM capabilities like mutual authentication, HMAC integrity, and role-based access control, the system maintains a high count of successful updates and drastically lowers the impact of adversary interference throughout the training rounds. The framework's dynamic client trust scoring methodology ensures that only trustworthy participants are permitted to influence model training, automatically decreasing the effect of suspicious or compromised clients in real time.

The Attack Detection and Resilience Module (ADRM) paired with robust aggregation (Krum) proves particularly efficient against poisoning and Byzantine attacks. The backdoor attack success rate for conventional FedAvg (no ADRM) soon grows and continues above 30%, but with ADRM enabled, this percentage is held close to zero or well below 30% throughout. Model test accuracy likewise supports the ADRM's value: under assault, Krum with ADRM continuously outperforms the baseline, offering greater and more stable accuracy even when faced with persistent adversarial activity. The ADRM's anomaly detection is efficient and lightweight, retaining high detection rates, rapid time-to-mitigation, and low extra delay in IIoT installations.

In practical IIoT settings, the Secure Aggregation Module (SAM) successfully guarantees privacy and dropout resilience. Aggregation remains robust up to a client dropout rate of 40%, after which success falls sharply, consistent with the theoretical threshold of the cryptographic scheme. This conclusion establishes a clear operational parameter for system design. The aggregated sum histogram affirms that the contribution of any single user to the global result is mathematically hidden; when one participant is removed, the sum's statistical distribution remains essentially unchanged, so privacy is preserved even if attackers observe multiple rounds or orchestrate targeted dropouts.

The protocol also provides efficient, privacy-preserving aggregation without sacrificing model utility: the aggregated model accuracy remains high and on par with non-secure baselines even during client loss and network instability. Large-scale trials with several rounds reveal that, with only basic encryption (TLS 1.2), global model accuracy quickly collapses, whereas conventional

SCPM improves stability yet remains vulnerable to advanced threats. Advanced SCPM maintains model accuracy above 70% and typically over 80–90% for several rounds, reflecting successful resistance against a spectrum of threats as well as controlled access, authorized updates, and rigorous integrity enforcement.

Collectively, these findings support the adoption of the framework’s tightly integrated modules including secure transport and protocol enforcement, privacy-preserving computation with Differential Privacy and Homomorphic Encryption, robust and dropout-tolerant aggregation, and real-time adversarial defense for practical, scalable, and trustworthy federated learning in industrial and critical infrastructure environments.

5 CONCLUSION

This thesis proposes a coherent and practical architecture for secure, privacy-preserving, and robust federated learning adapted to the demands of Industrial IoT contexts. By integrating modular components for secure communication, protocol enforcement, privacy preservation, robust aggregation, and real-time attack prevention, the system addresses both well-studied and new risks at every tier of the collaborative learning process. The suggested architecture demonstrates how enhanced command authentication, mutual TLS, and granular access control may successfully insulate IIoT control channels from manipulation and unwanted access. Combined with lightweight differential privacy, homomorphic encryption, and secure aggregation based on masking and secret sharing, the framework ensures individual client updates remain confidential, withstands significant client dropout, and maintains accuracy on par with centralized or non-secure baselines.

Empirical study through realistic assault scenarios indicates that the global model's integrity is sustained even under prolonged adversarial pressure, with defense modules preventing or mitigating model poisoning, Byzantine errors, and backdoor insertion. The results demonstrate that dynamic trust scoring, anomaly detection, and robust aggregation techniques prevent both targeted and dispersed threats without incurring undue latency or communication overhead communication. Privacy audits and resilience testing further prove that changes in federated aggregation are unrecognizable during user churn, hence defending against both direct exfiltration and indirect inference threats.

Altogether, our study proves that secure, attack-resilient, and privacy-aware federated learning is viable at scale across resource-constrained, unreliable industrial networks. The solution creates a platform for trustworthy IIoT deployments where sensitive operational data never reaches the edge, yet collaborative analytics and predictive modeling remain possible and resilient. Future work can enhance this framework with adaptive and intelligent auto-configuration, broader heterogeneous testbeds, and active protection against zero-day assaults, leading towards fully autonomous, safe, and privacy-compliant industrial machine learning.

6 REFERENCES

[1] Li, T. et al., "Byzantine-robust federated learning," 2023.

[2] Dwork, Cynthia, "Differential privacy: A survey of results," 2008.

[3] Febrianti Wibawa, Ferhat Ozgur Catak, Murat Kuzlu, Salih Sarp, Umit Cali,, ""Homomorphic Encryption and Federated Learning based Privacy-Preserving CNN Training: COVID-19 Detection Use-Case," 2022.

[4] Yichen Ren, Gu Li, Cheng Liu, Ru Chen, Bin Xu, and Jianing Wu, "Defense strategies toward model poisoning attacks in federated learning: A survey," 2024.

[5] Chen, Lian and Wang, Ming and Zhang, Hui and Zhang, Hao, "Privacy threats and countermeasures in federated learning for internet of things: A systematic review," 2024.

[6] Bonawitz, K. et al., "Practical secure aggregation for privacy-preserving machine learning," 2017.

[7] Rathee, Mayank and Shen, Conghao and Wagh, Sameer and Popa, Raluca Ada, "ELSA Secure Aggregation for Federated Learning with Malicious Actors," 2023.

[8] Li, X. et al, "Federated Learning with Differential Privacy and Homomorphic Encryption," 2023.

[9] Biggio, B. et al., "Poisoning attacks against machine learning systems," 2012.

[10] Shapsough, Shams and Aloul, Fadi and Zualkernan, Imran, "Securing Low-Resource Edge Devices for IoT Systems," 2018.

[11] Zhu, J., "Role-Based Access Control models for IoT Applications," 2023.

[12] M. Riazi, "Efficient HE for IIoT devices," 2024.

[13] Zhang, Y. et al., "Threshold secret sharing in federated learning," 2023.

[14] Bonawitz, K. et al., "Multi-round secure aggregation: theory and practice," 2022.

[15] Fang, Q. et al., "Real-time anomaly detection in federated learning," 2024.

[16] Q. Han, S. Yang, X. Ren, P. Zhao, C. Zhao and Y. Wang, ", "PCFed: Privacy-Enhanced and Communication-Efficient Federated Learning for Industrial IoTs," 2024.

[17] Christin Alex, Gisella Creado, Wesam Almobaidenn, Orieb Abu Alghanam, Maha , "A Comprehensive Survey for IoT Security Datasets Taxonomy, Classification and Machine Learning Mechanisms,," 2023.

[18] Bian Zhu, Ling Niu, "A privacy-preserving federated learning scheme with homomorphic encryption and edge computing".

[19] Dayane Reis; Jonathan Takeshita; Taeho Jung; Michael Niemier; Xiaobo Sharon Hu, "Computing-in-Memory for Performance and Energy-Efficient Homomorphic Encryption".

[20] Jie Fu, Yuan Hong, Xinpeng Ling, Leixia Wang, Xun Ran, Zhiyu Sun, Wendy Hui Wang, Zhili Chen, Yang Cao, "Differentially Private Federated Learning: A Systematic Review," 2024.

[21]] Lingjuan Lyu; Han Yu; Xingjun Ma; Chen Chen; Lichao Sun; Jun Zhao, "Privacy and Robustness in Federated Learning: Attacks and Defenses," 2023.

[22] A. a. F. M. a. C. A. a. V. M. Catalfamo, "Privacy-Preserving in Federated Learning: A Comparison between Differential Privacy and Homomorphic Encryption across Different Scenarios,," 2025.

[23] Shenghui Li, Edith Ngai,Thiemo Voigt, "Byzantine-Robust Aggregation in Federated Learning," 2021.

[24] Mohamad Mansouri, Melek Önen, Wafa Ben Jaballah, Mauro Conti, "SoK: Secure Aggregation Based on Cryptographic Schemes for," , 2023..

[25] Shamir, Adi, ""How to share a secret," Association for Computing Machinery, 1979.

[26] H. Brendan McMahan and Eider Moore and Daniel Ramage and Seth Hampson and Blais Agüera y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," 2023.

[27] Timothy Stevens, Christian Skalka,Christelle Vincent, John Ring, Samuel Clark, Joseph , "Efficient Differentially Private Secure Aggregation for Federated Learning," 2022.

[28] Ahmed Imteaj, Urmish Thakker, Shiqiang Wang, M. Hadi Amini, "A Survey on Federated Learning," IEEE INTERNET OF THINGS JOURNAL, 2022.

[29] A Ahmad, N Khan, N Zareen, S Ali, and A Shah, "Trust-based anomaly detection and mitigation in iot networks," Journal of Network and Computer Applications, 2024.

[30] L Chen, M Wang, and H Zhang, "Adversarial attacks on iot anomaly detection systems," IEEE International Conference on IoT, 2024 .

[31] Li Ding, Yali Zhao, and Haotian Wang, "Threshold-based anomaly detection for gradient inversion attacks in federated learning," 2024.

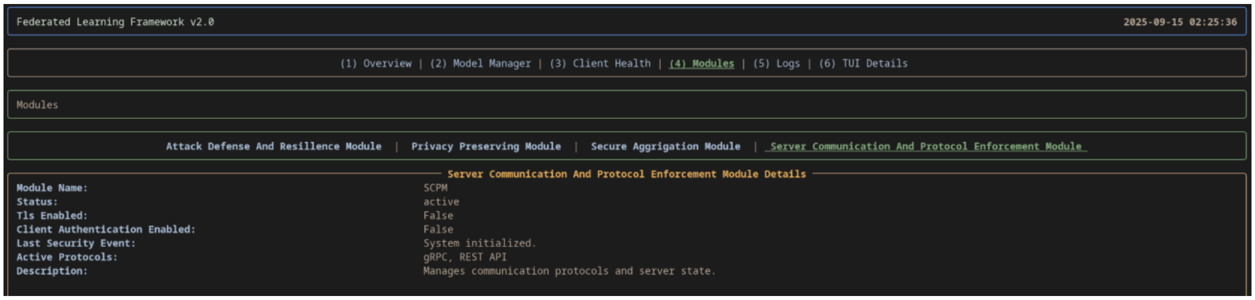
[32] Goutham Godavarthi, Pradip Gopinath, Soumya Sahu, and Suman Kumar, "Federated learning against byzantine attacks with multi-layer defense," 2024.

[33] Md Zarif Hossain, Ahmed Imteaj, and Abdur R Shahid, "Flamingo: Adaptive and resilient federated meta-learning against adversarial attack," 2024.

SUMMARY OF EACH STUDENTS CONTRIBUTION

Secure Communication and Protocol Enforcement Module (SCPM)

Core Responsibilities: Led the design and implementation of the framework's foundational C2 security architecture.



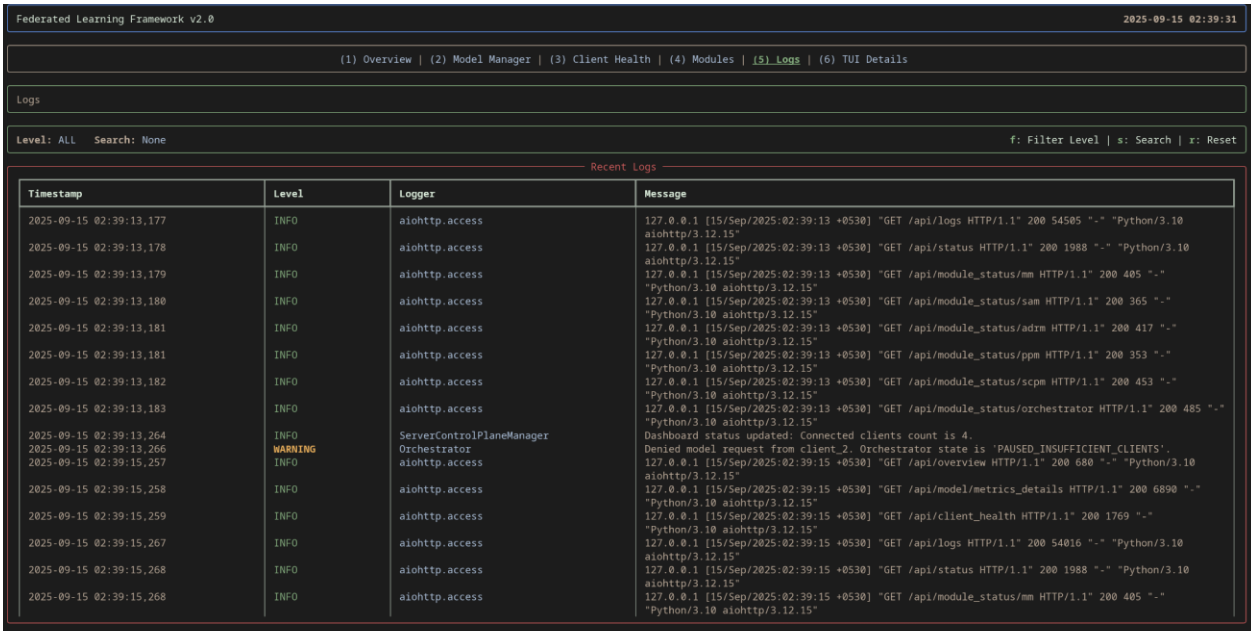
```
Federated Learning Framework v2.0                                                              2025-09-15 02:25:36

          (1) Overview | (2) Model Manager | (3) Client Health | (4) Modules | (5) Logs | (6) TUI Details

Modules

  Attack Defense And Resillence Module | Privacy Preserving Module | Secure Aggrigation Module | Server Communication And Protocol Enforcement Module

                          Server Communication And Protocol Enforcement Module Details
Module Name:                                   SCPM
Status:                                        active
Tls Enabled:                                   False
Client Authentication Enabled:                 False
Last Security Event:                           System initialized.
Active Protocols:                              gRPC, REST API
Description:                                   Manages communication protocols and server state.
```

Figure 17 : SCPM Status - TUI

Key Implementations:

- Secured all communication channels using TLS 1.3 with mutual authentication (mTLS) to ensure confidentiality and prevent unauthorized device connections or Man-in-the-Middle (MitM) attacks.
- Integrated HMAC-based integrity verification to protect all messages against tampering and replay attacks.
- Developed and enforced a Role-Based Access Control (RBAC) mechanism to ensure authenticated clients could only perform permitted actions.

Evaluation: Conducted tests to validate that the SCPM successfully blocks unauthenticated connections and rejects tampered messages , demonstrating its role in maintaining high global model accuracy in adversarial environments.



Federated Learning Framework v2.0 2025-09-15 02:39:31

(1) Overview | (2) Model Manager | (3) Client Health | (4) Modules | (5) Logs | (6) TUI Details

Logs

Level: ALL Search: None f: Filter Level | s: Search | r: Reset

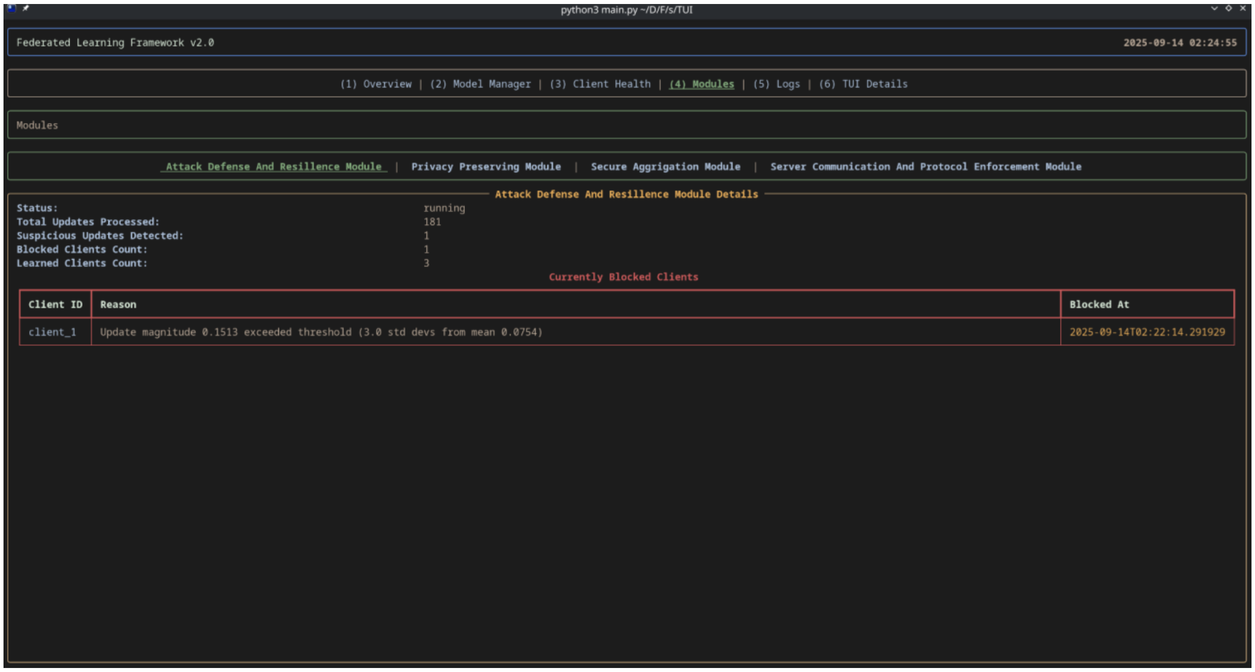
Recent Logs

Timestamp	Level	Logger	Message
2025-09-15 02:39:13,177	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/logs HTTP/1.1" 200 54505 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,178	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/status HTTP/1.1" 200 1988 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,179	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/mm HTTP/1.1" 200 405 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,180	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/sam HTTP/1.1" 200 365 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,181	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/adrm HTTP/1.1" 200 417 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,181	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/ppm HTTP/1.1" 200 353 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,182	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/scpm HTTP/1.1" 200 453 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,183	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/orchestrator HTTP/1.1" 200 485 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:13,264	INFO	ServerControlPlaneManager	Dashboard status updated: Connected clients count is 4.
2025-09-15 02:39:13,266	WARNING	Orchestrator	Denied model request from client_2. Orchestrator state is 'PAUSED_INSUFFICIENT_CLIENTS'.
2025-09-15 02:39:15,257	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/overview HTTP/1.1" 200 680 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:15,258	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/model/metrics_details HTTP/1.1" 200 6890 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:15,259	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/client_health HTTP/1.1" 200 1769 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:15,267	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/logs HTTP/1.1" 200 54016 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:15,268	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/status HTTP/1.1" 200 1988 "-" "Python/3.10 aiohttp/3.12.15"
2025-09-15 02:39:15,268	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/module_status/mm HTTP/1.1" 200 405 "-" "Python/3.10 aiohttp/3.12.15"

Figure 18 : Real Time Logger - TUI

Attack Detection and Resilience Module (ADRM)

Core Responsibilities: Engineered the framework's real-time active defense system to protect the global model's integrity from malicious clients.



python3 main.py ~/D/F/s/TUI

Federated Learning Framework v2.0 2025-09-14 02:24:55

(1) Overview | (2) Model Manager | (3) Client Health | (4) Modules | (5) Logs | (6) TUI Details
Modules
Attack Defense And Resillence Module | Privacy Preserving Module | Secure Aggrigation Module | Server Communication And Protocol Enforcement Module

Attack Defense And Resillence Module Details

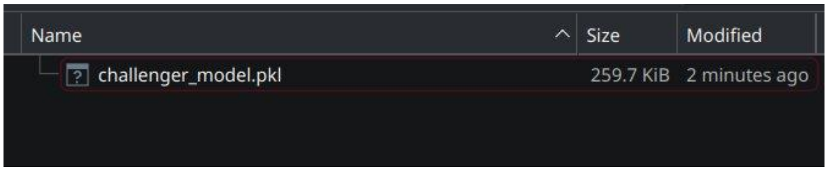
```
Status:                          running
Total Updates Processed:         181
Suspicious Updates Detected:     1
Blocked Clients Count:           1
Learned Clients Count:           3
```

Currently Blocked Clients

Client ID	Reason	Blocked At
client_1	Update magnitude 0.1513 exceeded threshold (3.0 std devs from mean 0.0754)	2025-09-14T02:22:14.291929

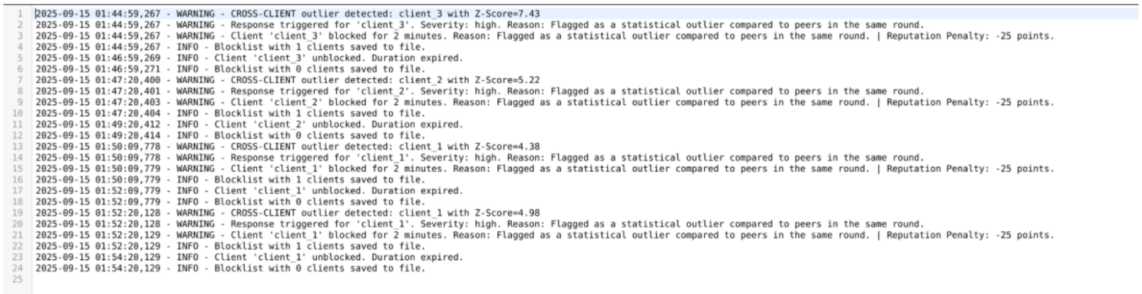
Figure 19 : ADRM Status - TUI

Key Implementations:



Name	Size	Modified
challenger_model.pkl	259.7 KiB	2 minutes ago

Figure 20 : ADRM ML Model



```
2025-09-15 01:44:59,267 - WARNING - CROSS-CLIENT outlier detected: client_3 with Z-Score=7.43
2025-09-15 01:44:59,267 - WARNING - Response triggered for 'client_3'. Severity: high. Reason: Flagged as a statistical outlier compared to peers in the same round.
2025-09-15 01:44:59,267 - WARNING - Client 'client_3' blocked for 2 minutes. Reason: Flagged as a statistical outlier compared to peers in the same round. | Reputation Penalty: -25 points.
2025-09-15 01:44:59,267 - INFO - Blocklist with 1 clients saved to file.
2025-09-15 01:46:59,269 - INFO - Client 'client_3' unblocked. Duration expired.
2025-09-15 01:46:59,271 - INFO - Blocklist with 0 clients saved to file.
2025-09-15 01:47:20,400 - WARNING - CROSS-CLIENT outlier detected: client_2 with Z-Score=5.22
2025-09-15 01:47:20,401 - WARNING - Response triggered for 'client_2'. Severity: high. Reason: Flagged as a statistical outlier compared to peers in the same round.
2025-09-15 01:47:20,403 - WARNING - Client 'client_2' blocked for 2 minutes. Reason: Flagged as a statistical outlier compared to peers in the same round. | Reputation Penalty: -25 points.
2025-09-15 01:47:20,404 - INFO - Blocklist with 1 clients saved to file.
2025-09-15 01:49:20,412 - INFO - Client 'client_2' unblocked. Duration expired.
2025-09-15 01:49:20,414 - INFO - Blocklist with 0 clients saved to file.
2025-09-15 01:50:09,778 - WARNING - CROSS-CLIENT outlier detected: client_1 with Z-Score=4.38
2025-09-15 01:50:09,778 - WARNING - Response triggered for 'client_1'. Severity: high. Reason: Flagged as a statistical outlier compared to peers in the same round.
2025-09-15 01:50:09,779 - WARNING - Client 'client_1' blocked for 2 minutes. Reason: Flagged as a statistical outlier compared to peers in the same round. | Reputation Penalty: -25 points.
2025-09-15 01:50:09,779 - INFO - Blocklist with 1 clients saved to file.
2025-09-15 01:52:09,779 - INFO - Client 'client_1' unblocked. Duration expired.
2025-09-15 01:52:09,779 - INFO - Blocklist with 0 clients saved to file.
2025-09-15 01:52:20,128 - WARNING - CROSS-CLIENT outlier detected: client_1 with Z-Score=4.98
2025-09-15 01:52:20,128 - WARNING - Response triggered for 'client_1'. Severity: high. Reason: Flagged as a statistical outlier compared to peers in the same round.
2025-09-15 01:52:20,129 - WARNING - Client 'client_1' blocked for 2 minutes. Reason: Flagged as a statistical outlier compared to peers in the same round. | Reputation Penalty: -25 points.
2025-09-15 01:52:20,129 - INFO - Blocklist with 1 clients saved to file.
2025-09-15 01:54:20,129 - INFO - Client 'client_1' unblocked. Duration expired.
2025-09-15 01:54:20,129 - INFO - Blocklist with 0 clients saved to file.
```

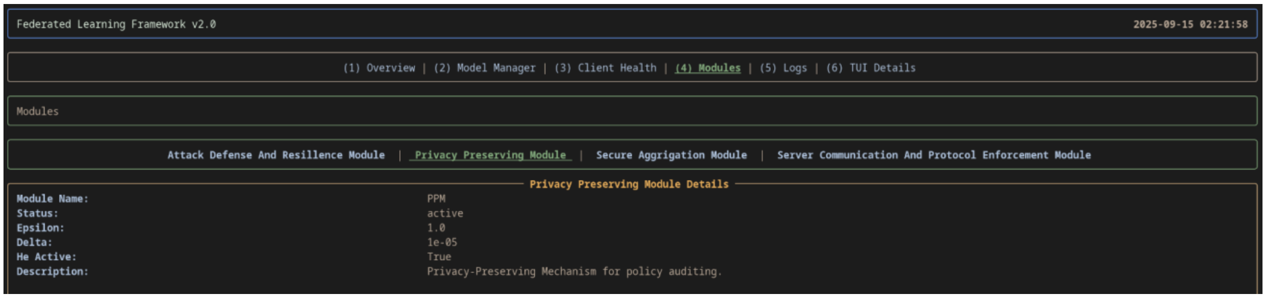
Figure 21 : ADRM Logs

- Developed a lightweight, server-side anomaly detection pipeline to intercept and score incoming client updates in real-time.
- Implemented an Isolation Forest model for its low computational overhead and high accuracy in detecting outlier updates.
- Integrated robust aggregation algorithms, specifically Krum, to work with the detection engine to mitigate attacks.
- Implemented automated mitigation techniques, including the ability to down-weight, reject, or quarantine malicious updates.

Evaluation: Tested the ADRM in scenarios with model poisoning and backdoor attacks , proving it maintained global model accuracy above 85% and kept backdoor success rates near 0% , significantly outperforming the baseline FedAvg model.

Privacy Preservation Module (PPM)

Core Responsibilities: Implemented the core privacy-enhancing techniques to protect the content of client model updates from being exposed.

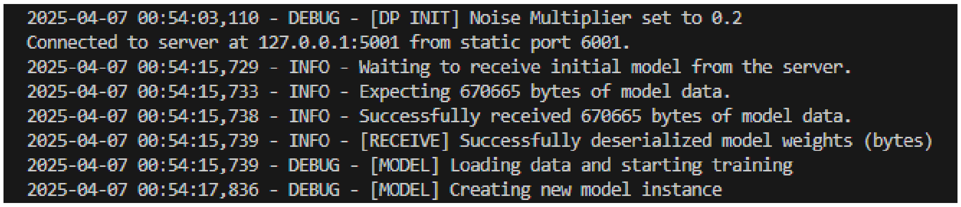


```
Federated Learning Framework v2.0                                                    2025-09-15 02:21:58

        (1) Overview | (2) Model Manager | (3) Client Health | (4) Modules | (5) Logs | (6) TUI Details

Modules

Attack Defense And Resillence Module | Privacy Preserving Module | Secure Aggrigation Module | Server Communication And Protocol Enforcement Module

---------------------------------- Privacy Preserving Module Details ----------------------------------
Module Name:                       PPM
Status:                            active
Epsilon:                           1.0
Delta:                             1e-05
He Active:                         True
Description:                       Privacy-Preserving Mechanism for policy auditing.
```

Figure 22 : PPM Status - TUI

Key Implementations:

- Incorporated Differential Privacy (DP) using the Opacus library, automating gradient clipping and the addition of calibrated Gaussian noise to client updates before transmission.

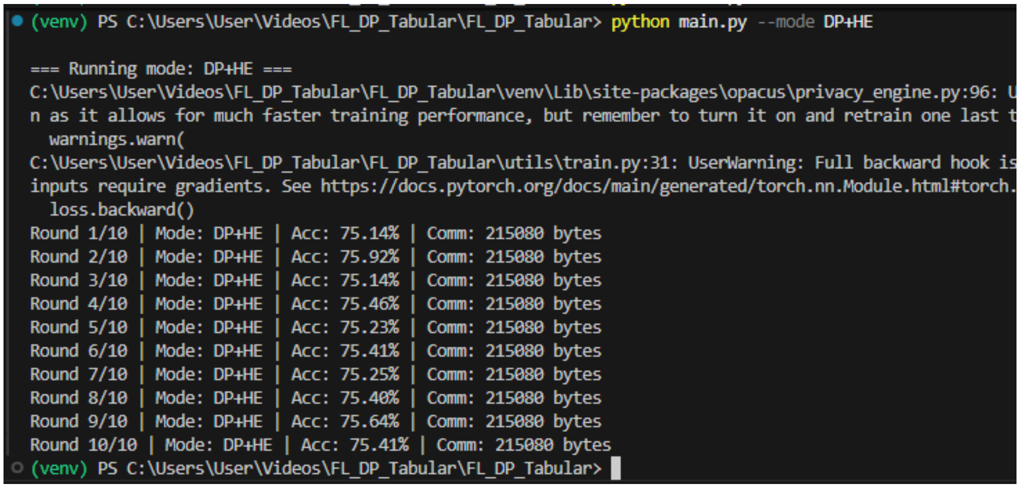


```
2025-04-07 00:54:03,110 - DEBUG - [DP INIT] Noise Multiplier set to 0.2
Connected to server at 127.0.0.1:5001 from static port 6001.
2025-04-07 00:54:15,729 - INFO - Waiting to receive initial model from the server.
2025-04-07 00:54:15,733 - INFO - Expecting 670665 bytes of model data.
2025-04-07 00:54:15,738 - INFO - Successfully received 670665 bytes of model data.
2025-04-07 00:54:15,739 - INFO - [RECEIVE] Successfully deserialized model weights (bytes)
2025-04-07 00:54:15,739 - DEBUG - [MODEL] Loading data and starting training
2025-04-07 00:54:17,836 - DEBUG - [MODEL] Creating new model instance
```

Figure 23 : DP Initialization Log

- Implemented Homomorphic Encryption (HE) using the Paillier cryptosystem, allowing the server to aggregate encrypted updates without possessing the private key.

Evaluation: Led the comparative analysis of the privacy-accuracy-efficiency trade-offs. This analysis quantified the performance of four distinct configurations (Baseline FL, FL+DP, FL+HE, and FL+DP+HE), showing the accuracy impact of DP ~65% accuracy) versus the communication overhead of HE (5-10x increase).

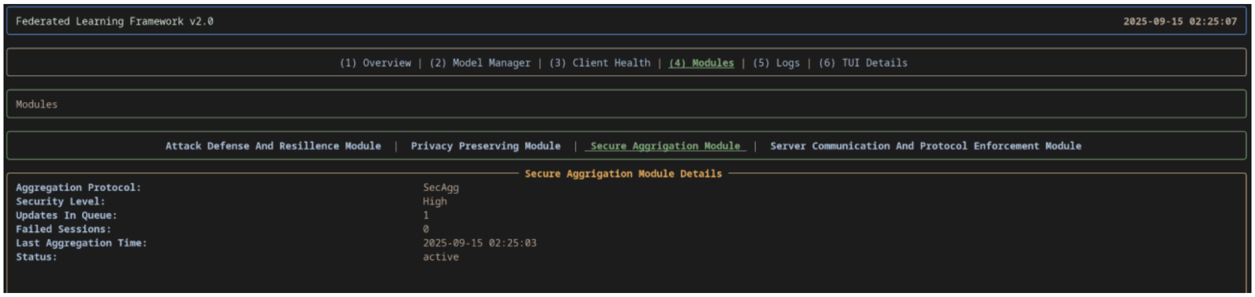


```
(venv) PS C:\Users\User\Videos\FL_DP_Tabular\FL_DP_Tabular> python main.py --mode DP+HE

=== Running mode: DP+HE ===
C:\Users\User\Videos\FL_DP_Tabular\FL_DP_Tabular\venv\Lib\site-packages\opacus\privacy_engine.py:96: U
n as it allows for much faster training performance, but remember to turn it on and retrain one last t
  warnings.warn(
C:\Users\User\Videos\FL_DP_Tabular\FL_DP_Tabular\utils\train.py:31: UserWarning: Full backward hook is
inputs require gradients. See https://docs.pytorch.org/docs/main/generated/torch.nn.Module.html#torch.
  loss.backward()
Round 1/10 | Mode: DP+HE | Acc: 75.14% | Comm: 215080 bytes
Round 2/10 | Mode: DP+HE | Acc: 75.92% | Comm: 215080 bytes
Round 3/10 | Mode: DP+HE | Acc: 75.14% | Comm: 215080 bytes
Round 4/10 | Mode: DP+HE | Acc: 75.46% | Comm: 215080 bytes
Round 5/10 | Mode: DP+HE | Acc: 75.23% | Comm: 215080 bytes
Round 6/10 | Mode: DP+HE | Acc: 75.41% | Comm: 215080 bytes
Round 7/10 | Mode: DP+HE | Acc: 75.25% | Comm: 215080 bytes
Round 8/10 | Mode: DP+HE | Acc: 75.40% | Comm: 215080 bytes
Round 9/10 | Mode: DP+HE | Acc: 75.64% | Comm: 215080 bytes
Round 10/10 | Mode: DP+HE | Acc: 75.41% | Comm: 215080 bytes
(venv) PS C:\Users\User\Videos\FL_DP_Tabular\FL_DP_Tabular>
```

Figure 24 : DP + HE Testing Results

Secure Aggregation Module (SAM)

Core Responsibilities: Designed and implemented the protocol to ensure the aggregation process was both privacy-preserving and resilient to client failures, which are common in IIoT networks.



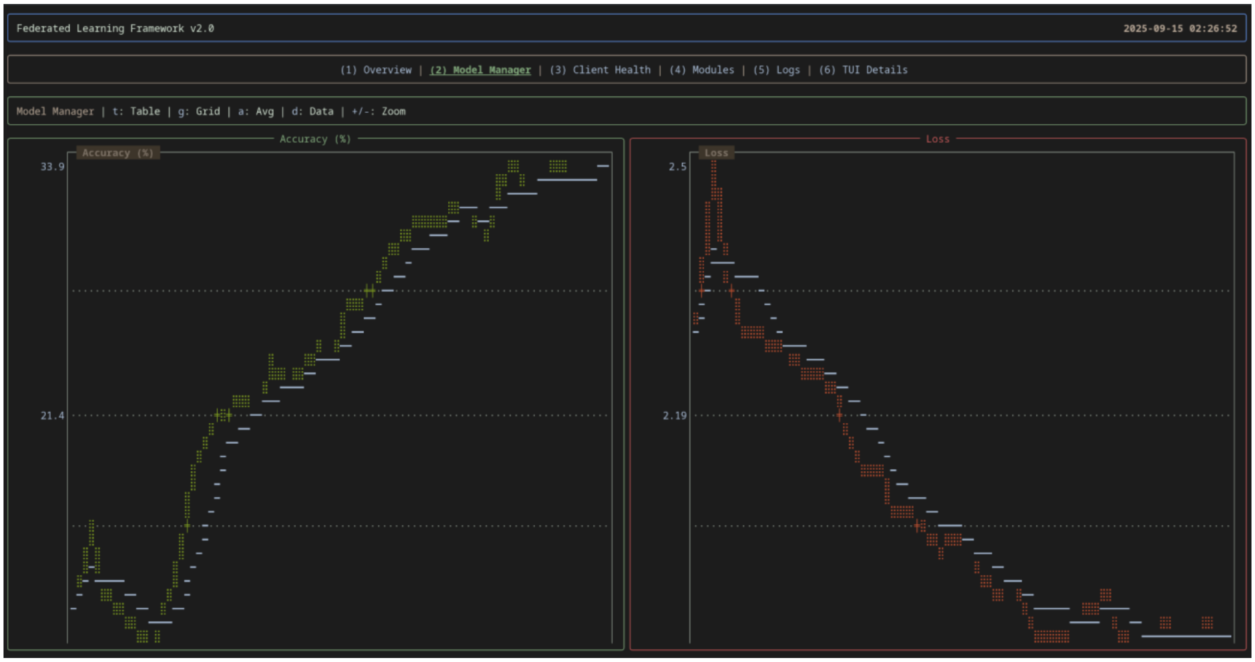
```
Federated Learning Framework v2.0                                                   2025-09-15 02:25:07

          (1) Overview | (2) Model Manager | (3) Client Health | (4) Modules | (5) Logs | (6) TUI Details

Modules

   Attack Defense And Resillence Module | Privacy Preserving Module | Secure Aggrigation Module | Server Communication And Protocol Enforcement Module

                                      Secure Aggrigation Module Details
Aggregation Protocol:                      SecAgg
Security Level:                            High
Updates In Queue:                          1
Failed Sessions:                           0
Last Aggregation Time:                     2025-09-15 02:25:03
Status:                                    active
```

Figure 25 : SAM Status - TUI

Key Implementations:

- Implemented pairwise masking, where clients use shared secrets to cryptographically hide their individual contributions from the server.
- Integrated Shamir's Secret Sharing (SSS) as a threshold-based recovery mechanism. This allows the server to reconstruct and remove the masks of dropped-out clients, ensuring the aggregation can be completed successfully.

Evaluation: Validated the protocol's robustness, achieving a 90% aggregation success rate with up to a 30% client dropout rate and identifying the theoretical failure point (~40% dropout). Also demonstrated the privacy guarantee by showing that the statistical distribution of aggregated sums remains unchanged when a single user is removed, proving individual contributions cannot be inferred.



Federated Learning Framework v2.0 2025-09-15 02:26:52

(1) Overview | (2) Model Manager | (3) Client Health | (4) Modules | (5) Logs | (6) TUI Details

Model Manager | t: Table | g: Grid | a: Avg | d: Data | +/-: Zoom

Figure 26 : Aggregation Process - TUI