REMINDER

REMINDER:

pRivacy-prEserving Machine LearnIng through secure managemeNt of Data's lifecyclE in distRibuted systems

Deliverable number: D1.2

Final requirements specification

chist-era

FWF Austrian Science Fund

UKRI Engineering and Physical Sciences Research Council

uefiscdi Unitatea Executivă pentru Finanțarea Învățământului Superior, a Cercetării, Dezvoltării și Inovării

AGENCIA ESTATAL DE INVESTIGACIÓN

Project Acronym:	REMINDER
Project Full Title:	pRivacy-prEserving Machine LearnIng through secure manageMeNt of Data's lifecyclE in distRibuted systems
Call:	Security and Privacy in Decentralised and Distributed Systems (SPiDDS). 2022
Grant Number:	PCI2023-145989-2
Project URL:	https://ants.inf.um.es/en/reminder
Editor:	UMU
Deliverable nature:	Report
Dissemination level:	Public
Delivery Date:	31/03/2025
Authors:	UMU, SIE, UWE, AIT

Table 1: Project details.

Abstract

Artificial Intelligence (AI) is increasingly deployed across diverse sectors to enable real-time, data-driven decision-making. While these advances offer substantial benefits, they also introduce pressing challenges related to data security, privacy, and regulatory compliance. The REMINDER project addresses these concerns by leveraging a novel Federated Learning (FL) framework designed to preserve confidentiality while accommodating resource-constrained devices at the network edge. Through privacy-enhancing techniques—such as Differential Privacy (DP), Homomorphic Encryption (HE), robust aggregation, and malicious client detection—REMINDER ensures that sensitive information remains secure throughout the data lifecycle. Additionally, the framework incorporates authentication protocols to verify legitimate participants, reducing the risk of adversarial tampering. This document provides an overview of REMINDER's edge-based architecture and its comprehensive validation in two representative domains: healthcare, where FL supports privacy-preserving collaborative diagnosis, and smart buildings, where occupant privacy and energy efficiency are improved through distributed intelligence. By integrating robust security measures with scalable FL operations, REMINDER demonstrates a practical path toward deploying AI-driven solutions without compromising data confidentiality or system integrity.

Table of Contents

List of Figures

List of Tables

1 Introduction

Federated Learning (FL) has emerged as a transformative approach for collaboratively training machine learning models across distributed and potentially sensitive data sources. Unlike traditional centralized paradigms that require transferring raw datasets to a single repository, FL enables data to remain locally on hospital servers, building management systems, or other edge devices—while only sharing model updates. This decentralized framework is particularly beneficial where privacy regulations (e.g., GDPR, HIPAA) prohibit direct data pooling, or where granular data exposure (such as energy consumption patterns in smart buildings) could jeopardize confidentiality. By avoiding the centralization of raw data, FL not only upholds privacy mandates but also leverages diverse, geographically dispersed insights to improve model robustness.

Despite these advantages, FL also introduces significant privacy and security challenges. The partial visibility of client updates can be exploited to perform inference attacks, allowing adversaries to reconstruct sensitive information from seemingly benign model parameters. In healthcare, even subtle gradients might inadvertently disclose patient medical records or conditions, risking non-compliance with strict regulations and undermining public trust. Meanwhile, in smart building applications, analyzing aggregated energy usage could reveal occupant habits or vulnerabilities if attackers correlate patterns from multiple sources. Moreover, malicious participants can degrade the overall performance by injecting poisoned updates or manipulating building controls to waste resources. As a result, robust protections—encompassing Differential Privacy (DP), Homomorphic Encryption (HE), digital signatures, and anomaly detection—are essential for any FL system seeking to safeguard data integrity and confidentiality.

In response, the REMINDER framework adopts FL principles within two key domains: healthcare and smart buildings. In healthcare, where tasks such as atrial fibrillation detection or aortic coarctation diagnosis rely on distributed clinical data, FL offers a compliant and privacy-aware solution for collaborative model training. For smart buildings, FL empowers facility managers to optimize energy usage based on local consumption patterns, preserving occupant privacy while maintaining secure control over infrastructure systems. Both scenarios underscore the necessity for a privacy-centric design that harmonizes edge intelligence with reliability and compliance.

To meet these needs, REMINDER integrates advanced privacy-preserving methods—such as DP, HE, robust aggregation mechanisms, and malicious client detection—into the federated pipeline. These enhancements counter adversarial threats (e.g., poisoning attacks) and guard against inference risks by obfuscating or encrypting data contributions. Equally important, the framework's modular architecture enables compatibility with heterogeneous deployments, including legacy systems, ensuring flexibility and scalability across diverse environments. By addressing these foundational requirements, REMINDER demonstrates how federated learning can be securely and effectively harnessed for next-generation healthcare and smart building solutions, ultimately enhancing data-driven outcomes without compromising stakeholder trust or regulatory obligations.

2 REMINDER Architecture

The REMINDER project presents a privacy-preserving and decentralized Federated Learning (FL) framework designed to meet the requirements established in this document, it will be explained in detail in Deliverable 2. This architecture ensures compliance with key principles, including:

Communication and Data Transmission – Secure, efficient, and authenticated data exchange across all system components.

Privacy and Security – Protection of sensitive information through data obfuscation, confidentiality-preserving mechanisms, and secure model aggregation.

Scalability and Interoperability – Seamless integration with diverse smart building infrastructures, including modern and legacy systems, while supporting large-scale deployments.

Federated Learning and Energy Optimization – Adaptive, robust FL models capable of optimizing energy consumption dynamically while maintaining system efficiency.

Regulatory Compliance and Auditability – Alignment with privacy regulations and the implementation of transparent monitoring and auditing mechanisms.

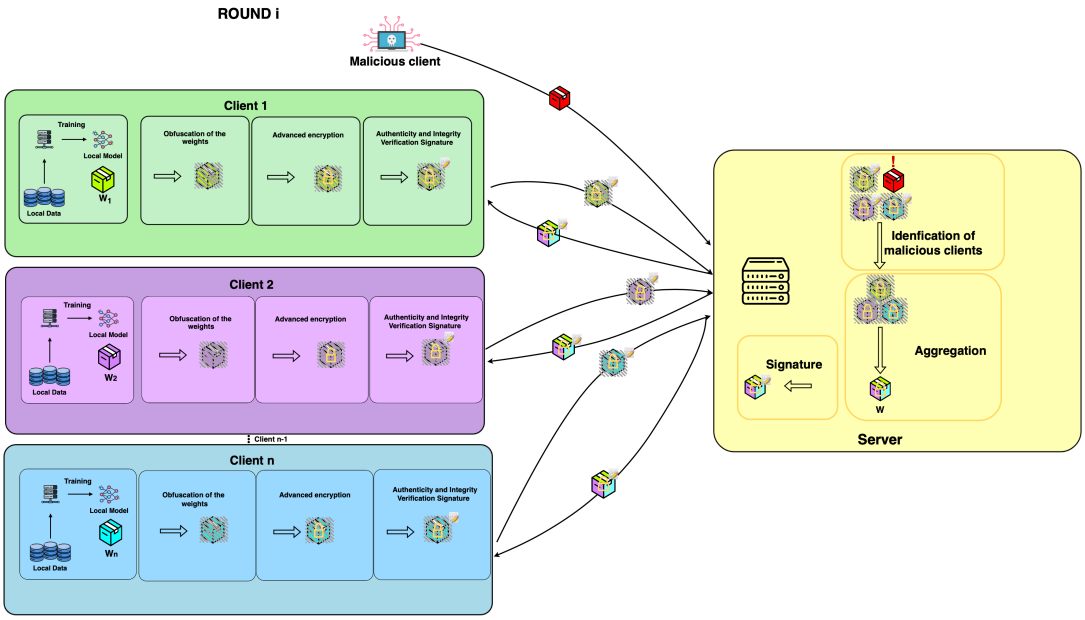

Figure 1: REMINDER Architecture

Our proposed architecture is presented in Fig. 1. In this case, during the communication between clients and server we add privacy-preserving techniques such us DP, encrypt methods and signature processes. And then, in the server, it is applied a different aggregation functions and frameworks for detecting malicious clients. In particular, the procedure of REMINDER will be:

(At clients' side)

1. We train the model in the "Trainer", a device which is technically capable of this task
2. After the training, the clients will apply DP techniques to obfuscate the weights.

3. The clients will cipher to add a higher degree of privacy. These updates are encrypted using a cipher. This ensures that even if intercepted by a malicious entity, it would be significantly more challenging to infer any sensitive information.
4. The model updates are signed to ensure their integrity and authenticity. The model updates are then sent to the server,

   (At servers' side)
5. The server performs a malicious client detector to make a first filter in order to remove as many malicious clients as possible.
6. The server aggregates all updates received from the clients using a robust aggregation function that discards malicious updates. The aggregation should be performed on encrypted data, ensuring that the server never has access to the actual weight values.

This process results in a global model that combines the knowledge from all legitimate clients, ensuring good performance while preserving the privacy of data from all sources. The server signs the updated global model to guarantee its provenance and integrity. The update is then sent to the known clients in its encrypted form, as the weights have not been decrypted at any stage of the process.

3 Challenges and Solutions

In this section, we introduce the challenges that REMINDER will face in its various use cases. Specifically, we address the vulnerabilities associated with poisoning attacks and inference attacks, as well as the strategic reduction in the number of clients to mitigate computational costs. These challenges represent critical hurdles in ensuring the robustness, privacy, and efficiency of the system in real-world applications.

3.1 Poisoning attacks

3.1.1 Definition

Poisoning attacks in federated learning (FL) are adversarial strategies designed to disrupt the training process through the introduction of maliciously modified data or model weights, aiming to degrade model performance [1]. These attacks generally manifest in two primary forms: data poisoning attacks and local model poisoning attacks. Data poisoning attacks involve adversaries deliberately altering the training data to manipulate the resulting model's predictions toward specific malicious goals or to reduce its accuracy. Within this category, two prominent types of attacks are Sybil and backdoor attacks. Sybil attacks consist of malicious participants creating multiple false identities, each contributing poisoned data to amplify the harmful effects. In contrast, backdoor attacks exploit FL's distributed structure by embedding adversarial triggers into subsets of training data across various participants [2].

Furthermore, data poisoning attacks can be classified according to the subtlety of their implementation: clean-label and dirty-label attacks [3]. Clean-label attacks subtly modify input samples without altering their corresponding labels, thus making detection challenging. Conversely, dirty-label attacks, such as label flipping, involve altering the labels assigned to the input samples—often significantly undermining model accuracy due to introduced inconsistencies. Label flipping specifically refers to randomly assigning incorrect labels to data points

without modifying the original data itself, posing a considerable threat due to its simplicity and effectiveness [4].

Local model poisoning attacks represent another significant class of adversarial techniques, where attackers modify the locally computed weights transmitted to the server rather than manipulating the dataset directly. These attacks include targeted and untargeted poisoning approaches [1]. Untargeted poisoning aims at broadly deteriorating global model accuracy through arbitrary corruptions, while targeted poisoning strategically influences model predictions for specific data points, preserving general performance on other benign samples. Untargeted attacks are often harder to detect and pose considerable risks due to their generalized detrimental effects [5]. Notably, empirical studies suggest local model poisoning might cause more severe damage compared to pure data poisoning attacks [6].

Moreover, untargeted poisoning attacks can involve adversarial collaboration among malicious clients. Such collaborative approaches entail participants independently training their models, averaging their resulting updates, and subsequently applying calculated perturbations to these combined weights, intentionally compromising the aggregated global model [7]. Prominent examples include the LIE (Little Is Enough) attack, where attackers add controlled small noise values to their averaged updates [8], and STATOPT (Static Optimization), which computes a fixed malicious direction to systematically degrade model performance [9]. Additionally, more sophisticated methods like min-max and min-sum attacks seek to maximize distances between the malicious and benign model updates, ensuring maximum disruption while evading robust aggregation defenses. Specifically, min-max optimizes the perturbation to increase the maximum distance without detection, whereas min-sum targets maximizing the summed distances among malicious updates [7].

Collectively, these poisoning attack methods pose significant threats to the integrity, reliability, and robustness of federated learning systems, underscoring the necessity for rigorous research and development of effective defensive strategies to counteract these adversarial threats.

3.1.2 Countermeasures

In this section, we discuss various defense strategies aimed at enhancing the security, integrity, and privacy of FL systems against adversarial threats such as poisoning attacks. Specifically, we examine robust aggregation functions that protect the global model by mitigating the influence of malicious or faulty client updates. Additionally, we explore mechanisms for malicious client detection, which proactively identify and exclude adversarial contributions, thus improving overall aggregation effectiveness. Finally, we highlight cryptographic signature schemes, which are essential for verifying data authenticity and maintaining the integrity of client updates through secure authentication protocols. These defensive approaches collectively form a comprehensive strategy to ensure resilient and trustworthy federated learning environments.

- **A Robust Aggregation Function**: In FL, the aggregation function plays a critical role in combining the knowledge learned by clients into a global model. A robust aggregation function not only merges updates but also mitigates the impact of malicious or faulty clients, ensuring the integrity of the global model. Robust aggregation enhances privacy by reducing the influence of outliers or adversarial updates, which could potentially leak sensitive information about other clients. Examples of these functions are the median [10], Krumm [11], and FedRDF [12].
- **Malicious clients detector**: Robust aggregation functions are an effective way to mitigate the impact of malicious clients. However, the performance of these functions depends

heavily on the number of malicious clients. Additionally, if there are not malicious clients, their performance is lower than FedAvg with no malicious clients. Hence, some scientists have researched methods for detecting which clients are sending malicious weights and removing them from the aggregation. These techniques are based mainly on clustering techniques that can classify between malign and benign clients.

- **Signature schemes**: Signature schemes are vital cryptographic tools for ensuring data authenticity and integrity in distributed systems like FL. Using private keys to generate unique digital signatures, these schemes verify data origins and prevent tampering via public-key cryptography [13, 14].

3.2 Inference attacks

3.2.1 Definition

Inference attacks in federated learning (FL) are adversarial strategies designed to extract sensitive information from shared model updates or the global model parameters, without directly accessing the original training data. These attacks exploit gradients or parameters exchanged during training to infer confidential characteristics of the underlying data. The three main types are: Model inversion, membership inference, and reconstruction attacks.

Model inversion attacks aim to reconstruct original input data from model outputs or gradients, leveraging knowledge of the model architecture and parameters [15]. Attackers attempt to solve an optimization problem to approximate input samples from observed outputs, typically through minimizing a loss function and a regularization term to enforce realistic data reconstructions. This process commonly employs gradient descent, iteratively adjusting inputs to reduce discrepancies between observed and reconstructed outputs. Model inversion is especially concerning in overfitted models, which may inadvertently memorize specific training samples, significantly threatening privacy in applications like facial recognition or medical diagnosis systems [16].

Membership inference attacks attempt to determine whether a particular data point was included in the model's training dataset [17, 18]. These attacks exploit differences in model behavior, such as confidence scores, loss values, or gradient norms, when evaluated on training versus non-training data points. Adversaries commonly utilize thresholds on confidence scores to classify input samples as either members or non-members of the training set. Membership inference presents severe privacy risks, notably in sensitive areas like healthcare and finance, as confirming membership implicitly reveals sensitive personal information.

Reconstruction attacks seek to reconstruct entire input datasets or their statistical properties directly from gradients or updates exchanged during FL training [19]. Unlike model inversion, which typically targets individual data points, reconstruction attacks exploit shared gradients from local clients, iteratively matching dummy inputs to observed gradients through optimization. By minimizing the difference between the observed gradients and those computed from candidate inputs, adversaries can reconstruct sensitive training data at scale. Reconstruction attacks thus pose a substantial privacy threat in collaborative environments, potentially compromising sensitive personal information shared among FL participants, such as medical records in healthcare applications.

Collectively, these inference attacks underscore significant vulnerabilities within federated learning systems, emphasizing the urgent need for robust privacy-preserving mechanisms to ensure

the confidentiality and integrity of participant data.

3.2.2 Countermeasures

In this section, we introduce key privacy-enhancing techniques employed within FL frameworks to safeguard sensitive participant data against inference threats and unauthorized access. Specifically, we examine Differential Privacy (DP), which introduces controlled noise into model updates to protect individual data contributions and ensure compliance with privacy regulations. Additionally, we explore Homomorphic Encryption (HE), a cryptographic approach enabling secure computations directly on encrypted data. Within HE, we distinguish between Full Homomorphic Encryption (FHE), which supports arbitrary encrypted computations at the cost of computational complexity, and Partial Homomorphic Encryption (PHE), offering efficiency but limited computational capabilities. Together, these methods provide essential privacy guarantees necessary for secure and compliant federated learning environments.

- **Differential Privacy (DP):** The data security of FL is usually supplemented with DP to further protect potentially sensitive training data [20]. DP introduces controlled noise into model updates, obscuring individual data contributions while retaining overall model accuracy. This ensures that adversaries cannot deduce specific information about any single data point, complying with GDPR standards. DP [21] is usually considered in the scope of FL settings due to the stringent communication requirements of other privacy-preserving approaches, such as secure multiparty computation [22].
- **Homomorphic Encryption (HE):** Envisioned by Rivest et al. in 1978 [23], HE allows operations to be performed on encrypted data without decryption, ensuring privacy. In FL, this enables data owners to encrypt their data before sending it to the server, which can process ciphertexts without accessing plaintext.
  - **Full Homomorphic Encryption (FHE):** It wasn't until 2009 when researchers realized that FHE was possible [24]. FHE supports arbitrary computations on encrypted data, including addition and multiplication, while keeping plaintext secure. Its versatility makes it ideal for FL use cases like robust aggregation operations. However, FHE is computationally intensive due to noise management and bootstrapping requirements [25]. Despite these challenges, advancements in open-source frameworks (e.g., HElib[1], OpenFHE[2]) and hardware acceleration initiatives (e.g., DARPA[3], Intel[4]) continue to improve its practicality [26].
  - **Partial Homomorphic Encryption (PHE):** PHE, a simpler alternative, supports single operations like addition or multiplication, offering greater efficiency than FHE in specific scenarios [27]. For example, the Paillier cryptosystem enables secure aggregation in FL [28]. However, PHE's limited scope makes it unsuitable for complex computations like the ones required by most robust aggregation functions [29].

[1] https://github.com/homenc/HElib
[2] https://www.openfhe.org/
[3] https://www.darpa.mil//news-events/2021-03-08
[4] https://dualitytech.com/partners/intel/

4 Use case 1 - eHealth

4.1 Context

Recent advances in computing and data storage have propelled innovations in healthcare, improving patient outcomes and contributing to rising global life expectancy [30]. Strict privacy regulations like the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA) limit the exchange of sensitive biomedical data, making it difficult to assemble large-scale datasets for AI research. FL, as introduced by Konecny et al. [31], addresses these challenges by training models across decentralized nodes without transferring raw patient data. Despite its benefits, FL is susceptible to adversarial interference (e.g., data or model poisoning) and privacy attacks like Membership Inference [32], Property Inference [33], and Data Reconstruction [34].

The REMINDER framework bolsters FL security using digital signatures to verify model updates, FFT-based aggregation to filter out malicious contributions, and differential privacy to safeguard sensitive information. Two healthcare applications demonstrate REMINDER's practicality: arrhythmia detection using wearable sensors to identify atrial fibrillation, and aortic coarctation diagnosis, where deep learning accelerates pressure-drop prediction. These use cases underscore the need for robust privacy protections to balance data-driven innovation with stringent regulatory requirements.

4.2 Requirements

Building on the core idea of enabling secure, privacy-preserving machine learning across multiple institutions, the following requirements integrate both the foundational FL features and the newly introduced techniques—Differential Privacy (DP), Homomorphic Encryption (HE), robust aggregation, malicious client detection, and signature schemes. These additions ensure comprehensive protection against inference attacks, adversarial manipulations, and unauthorized data access.

4.2.1 Functional Requirements

Federated Learning Setup

- The system must support distributed model training across multiple hospital nodes.
- Each node must train a local model using its own data without exporting any raw data.
- A central server must aggregate local model updates to produce a global model.
- The aggregation method should support partial participation, allowing nodes to contribute asynchronously.

Differential Privacy (DP)

- *Noise Generation and Budget Tracking:* In clinical scenarios such as arrhythmia detection, where ECG data from wearable sensors is collected in real time, the system must implement a noise generator that injects privacy-preserving noise into gradients or model parameters. Configurable parameters (e.g., scale, distribution of noise) should ensure compliance with a specific privacy budget ((ε, δ)), while the overhead remains minimal to support continuous data streams.

- *Computational Overhead and Resource Allocation:* For large-scale hospital networks or real-time monitoring of atrial fibrillation, sufficient CPU/GPU resources must be provisioned at each round of model training. Intensive DP operations—especially for high-dimensional data, such as multi-channel ECG signals—require efficient memory management to avoid compromising inference or feedback latency.
- *Automated Privacy Accounting:* The system should log the privacy budget spent per round (e.g., via Rényi Differential Privacy or other accounting methods), ensuring it remains within permissible thresholds. This is especially critical in sensitive medical applications like aortic coarctation diagnosis, where comprehensive patient data may span large anatomical measurements or imaging features.

Homomorphic Encryption (HE)

- *Cryptographic Library Support:* In healthcare deployments where wearable devices or clinical workstations encrypt patient data (e.g., ECGs or anatomical measurements) before transmission, the system must integrate proven HE libraries (e.g., HElib, Palisade, or OpenFHE) capable of handling large numeric vectors and supporting secure arithmetic operations.
- *Key Management and Distribution:* Hospitals or cardiology units must securely manage keys, ensuring only authorized entities possess the corresponding decryption capabilities. For arrhythmia monitoring, wearables could be provisioned with public keys, while private keys remain within a trusted clinical environment.
- **Full Homomorphic Encryption (FHE):**
  - *Arbitrary Computations at Scale:* Ideal for robust patient-specific modeling (e.g., advanced pressure-drop estimations for aortic coarctation), but requires significant compute resources for noise management and bootstrapping. Clinics may deploy specialized hardware (multi-core CPUs, GPUs, or FPGAs) to handle the extra overhead of continuous encrypted updates.
  - *Concurrency and Parallelization:* Parallelizable routines (e.g., for matrix multiplication under encryption) are essential for scaling to multi-hospital collaborations with large model sizes and many simultaneous updates.
- **Partial Homomorphic Encryption (PHE):**
  - *Targeted Operations (Addition or Multiplication):* Suitable for simpler tasks like secure sum aggregation of wearable data or partial computations in real-time ECG analysis. Significantly less overhead than FHE, but limited in the types of operations supported.
  - *Reduced Overhead:* Particularly beneficial for smaller, resource-constrained healthcare facilities or monitoring devices with limited computational capacity. However, more complex diagnostic calculations (e.g., advanced anomaly detection for aortic coarctation) may require fallback to other encryption modes or a trusted execution environment.

Robust Aggregation Function

- *Efficient Implementation:* Hospitals may collaborate to train models detecting arrhythmias or estimating hemodynamic parameters. The aggregator server must handle robust aggregation algorithms (e.g., median [10], Krumm [11], or FedRDF [12]) efficiently, even with thousands of concurrent updates from wearables or imaging devices.

- *Scalability and Parallelization:* Parallel data structures and multi-threaded computations are critical for prompt feedback, especially in real-time ECG monitoring (arrhythmia detection) or CFD-based calculations for coarctation diagnosis. Latency bottlenecks can negatively impact both clinical decision timelines and overall quality of service.
- *Load Balancing:* If numerous hospitals and outpatient clinics join the collaboration, multiple aggregator instances or load-balancing strategies should be adopted to maintain consistent performance and minimize queuing delays.

Malicious Client Detection

- *Clustering and Anomaly Detection:* Techniques that compare parameter distributions or gradient norms are essential for identifying compromised nodes (e.g., manipulated ECG feeds). This detection can operate online (per training round) or in scheduled batches without significantly delaying global updates.
- *On-The-Fly vs. Batch Processing:* Real-time detection is preferred in life-critical scenarios (e.g., arrhythmia monitoring) to promptly isolate suspicious updates. However, batch-based approaches can be used in post-hoc analyses for less time-sensitive diagnoses (e.g., aortic coarctation planning), balancing computational overhead and response time.
- *Adaptive Thresholds and Policies:* The system should enable dynamic tuning of sensitivity thresholds, reflecting varying data qualities or participation rates across different healthcare providers.

Signature Schemes

- *Key Storage and Management:* Private keys must be stored in secure hospital environments (e.g., hardware security modules) to ensure that only authorized devices (wearables, hospital servers) can generate valid signatures.
- *High-Throughput Verification:* When hundreds or thousands of ECG streams upload encrypted updates every minute, the aggregator must efficiently verify digital signatures (e.g., ECDSA) to avoid training slowdowns or data backlogs.
- *Tamper-Proof Logging:* Cryptographic hashing or secure logging ensures traceability of each update, vital for audits under stringent healthcare regulations (GDPR, HIPAA). This logging should be efficiently implemented so as not to overload storage systems or hamper real-time analysis needs.

By tailoring these advanced privacy and security requirements to the specific needs of arrhythmia detection and aortic coarctation diagnosis, healthcare institutions can implement federated learning frameworks that balance patient data confidentiality with state-of-the-art diagnostic performance. This alignment of computational capabilities, privacy-preserving techniques, and robust defense strategies enables clinicians to deliver timely and accurate interventions without compromising on regulatory compliance or patient safety.

4.2.2 Non-Functional Requirements

Performance

- Privacy-preservation techniques (DP, HE) and robust aggregation must preserve model accuracy while mitigating privacy leaks and adversarial risks.

- Cryptographic operations (e.g., FHE) should be optimized or selectively employed to minimize computational overhead and latency.
- Aggregation latency should remain low enough to support near real-time training requirements.

Scalability

- The architecture must accommodate dynamic node addition or removal without retraining the entire model.
- As data volumes and model complexity grow, aggregation algorithms (including HE-based or robust schemes) must scale efficiently.

Reliability

- The system must implement fault-tolerance, allowing federated training to continue even if some nodes drop out or send corrupted updates.
- Checkpointing and state-saving mechanisms must be enabled for seamless model recovery in case of system failures.

Security

- Nodes must authenticate with the central server using digital signatures to prevent unauthorized or malicious participation.
- Encrypted logs must be maintained to track model updates, aggregations, and access requests, while ensuring tamper-evident audit trails.

Privacy

- Privacy-preserving audit trails must be maintained without revealing sensitive patient data.
- DP must be configurable (privacy budget tuning) to balance privacy protection and model accuracy.
- When encryption is employed, plaintext data should remain inaccessible to all unprivileged entities, including the aggregator.

Regulatory and Compliance Requirements

- The system must comply with GDPR, HIPAA, and other relevant regulations by preventing any sharing of raw patient data outside secure hospital environments.
- Patient data must remain anonymized at all stages of training, aggregation, and encrypted communication.

Incorporating Differential Privacy, Homomorphic Encryption, robust aggregation, malicious client detection, and signature schemes fortifies federated learning systems against modern security and privacy threats. This comprehensive approach balances strict regulatory compliance with effective collaborative modeling, fostering innovation and improving patient care without compromising data confidentiality or model integrity.

5 Use case 2 - Smart Buildings

5.1 Context

The use of IoT devices in smart buildings, such as the Pleiades building at the University of Murcia, enables more efficient energy management by constantly monitoring environmental parameters (e.g., temperature, humidity, occupancy) and adjusting HVAC operations accordingly. However, collecting detailed energy usage data raises privacy concerns, as consumption patterns can reveal sensitive occupant information (e.g., daily routines, activities) when correlated with external sources [35].

FL offers a privacy-preserving solution by training models locally on building data and only exchanging aggregated updates instead of raw measurements [36]. This mitigates privacy risks while maintaining the benefits of collaborative learning, even in heterogeneous environments where buildings differ in sensor coverage. Federated Transfer Learning (FTL) further addresses data imbalance issues, transferring knowledge from well-instrumented buildings to those with limited IoT infrastructure.

In the Pleiades Building Management System (BMS), diverse devices (smart meters, HVAC controllers, Z-Wave sensors) continuously capture data in JSON format, while an asynchronous communication strategy minimizes latency. External environmental data from sources such as IMIDA and SIAM augments predictive models. Despite these technical advancements, legacy BMS protocols (e.g., Modbus, BACnet) pose integration challenges, prompting an interoperability layer to ensure older systems can also benefit from FL insights [37].

Finally, the REMINDER framework mitigates privacy and security risks by implementing digital signatures for authenticated updates, FFT-based robust aggregation (FedRDF) to filter malicious contributions, and differential privacy to obscure individual building data contributions. Such measures address the critical issue of inferring occupant behaviors from energy consumption, ensuring occupant confidentiality and safeguarding buildings from potential adversarial threats

5.2 Requirements

The following requirements integrate additional privacy and security modules into the REMINDER framework's Smart Buildings use case. These enhancements address diverse building infrastructures and ensure secure, interoperable FL for energy management.

5.2.1 Communication and Data Transmission

- **Secure and Encrypted Channels:** All communication between IoT devices, edge servers, and aggregator(s) must occur over secure protocols (e.g., TLS/SSL). If Homomorphic Encryption (HE) is employed, encrypted updates must be transmitted and processed efficiently to minimize overhead.
- **Authentication and Authorization:** IoT devices and edge nodes must authenticate (e.g., via digital signatures or PKI-based credentials) before transmitting data or model updates, ensuring only authorized participants contribute to federated training.
- **Low-Latency Data Flows:** Near real-time performance is necessary for both energy

optimization and FL updates. Cryptographic operations (HE, Differential Privacy) should be optimized to avoid prohibitive network delays.

- **Fault Tolerance:** Communication protocols must tolerate network disruptions. Nodes should buffer updates locally and synchronize asynchronously once connectivity resumes.
- **Data Consistency:** Mechanisms must detect and correct corrupted or inconsistent data packets (e.g., malformed ciphertexts) arising from transmission errors or device failures.

5.2.2 Privacy and Security

- **Differential Privacy (DP):** A system-wide mechanism must inject controlled noise into gradients or model updates under a configurable privacy budget, ensuring no individual building's patterns can be reverse-engineered.
- **Homomorphic Encryption (HE):** Partial (PHE) or Full (FHE) Homomorphic Encryption should be supported to enable secure computations on encrypted data, particularly when stakeholders demand higher confidentiality (e.g., occupant-level usage).
- **Robust Aggregation and Malicious Client Detection:** The aggregator must implement anomaly detection or clustering-based filtering (e.g., FFT-based robust aggregation, FedRDF). Compromised or adversarial nodes should be quarantined to prevent model poisoning.
- **Local Data Protection:** Edge nodes and IoT devices must securely store local data (e.g., on encrypted drives or secure hardware) to prevent unauthorized local access.
- **Signature Schemes:** Digital signatures or similar primitives must validate authenticity of all updates, preventing unauthorized or spoofed contributions to federated training.
- **Multi-Stakeholder Collaboration:** Privacy-preserving mechanisms must allow different organizations (building owners, facility managers) to collaborate without exposing raw or sensitive building data.
- **Compromised Node Mitigation:** The system should detect, isolate, and, if feasible, remediate nodes suspected of malicious activity until they are reauthorized or sanitized.

5.2.3 Scalability and Interoperability

- **Expandable FL Architecture:** As additional buildings and IoT endpoints join the federation, HE, DP, and robust aggregation modules must scale with no critical performance degradation.
- **Minimal Communication Overhead:** Large ciphertexts (due to HE) and DP noise can inflate data payloads. Protocols must be optimized (e.g., ciphertext compression) to conserve bandwidth.
- **Legacy BMS Integration:** The framework should include an interoperability layer mapping legacy protocols (Modbus, BACnet) to secure, FL-compatible channels, avoiding major infrastructure overhauls.
- **Modular and Extensible Interfaces:** New hardware components or protocols must integrate seamlessly, preserving existing security and FL capabilities with minimal reconfiguration.

- **Heterogeneous Hardware Compatibility:** The solution must operate effectively across edge gateways (e.g., Raspberry Pi) and high-performance servers (e.g., with GPU or FPGA accelerators for FHE).
- **Load Balancing and Distributed Processing:** To prevent bottlenecks, cryptographic operations and federated training tasks should be parallelized, distributing compute loads across the network.
- **Adaptive Energy Policies:** The system must allow configurable strategies for handling diverse regulatory or operational constraints (e.g., geographic or occupant preferences).

5.2.4 Federated Learning and Energy Optimization

- **Dynamic Adaptation:** FL models must adapt to changes in occupancy, weather, and energy demands, supporting continuous or near-continuous online learning.
- **Heterogeneous Sensor Participation:** Buildings with limited instrumentation can still contribute to FL; Federated Transfer Learning (FTL) or partial updates can transfer knowledge from data-rich sites.
- **Resilient Aggregation:** Aggregation functions (e.g., FedRDF, median-based) must resist adversarial updates, preserving model integrity even under high node diversity.
- **Continuous Learning Lifecycle:** The system should track and respond to model drift (e.g., occupant behavior shifts), adjusting FL hyperparameters or re-initializing partial training when needed.
- **Energy-Efficient Model Updates:** DP noise insertion and encryption should be optimized for resource-constrained devices, minimizing added CPU/GPU usage and ensuring sustainability.
- **Cross-Building Collaboration:** Aggregation strategies must fuse insights from geographically dispersed sites without raw data exchange, enabling large-scale improvements in energy efficiency.
- **Predictive Analytics:** FL models should incorporate external data (e.g., weather forecasts, occupancy trends) to anticipate demand peaks and proactively adjust building systems.

5.2.5 Regulatory Compliance and Auditability

- **Regulatory Adherence:** All data handling must follow GDPR, HIPAA (where applicable), and building-specific regulations for data protection and occupant privacy.
- **Auditable Secure Logs:** System actions (updates, detected anomalies, aggregator decisions) must be recorded in tamper-evident logs (e.g., blockchain-based or cryptographically hashed), supporting post-hoc analysis.
- **Clear Accountability Framework:** Building owners, facility managers, and IoT vendors should have well-defined responsibilities in maintaining data integrity, privacy, and security checks.

- **Explainability and Interpretability:** FL models making autonomous energy decisions (e.g., HVAC control) must allow stakeholders to examine and justify outcomes for transparency.
- **Ongoing Compliance:** Mechanisms (e.g., privacy budget accounting, data retention limits) must continually monitor and enforce regulatory thresholds, adjusting automatically if non-compliance risks are detected.

By addressing these requirements in communication, privacy, security, scalability, FL-driven energy optimization, and regulatory compliance, the REMINDER framework ensures that smart buildings can leverage federated learning without compromising occupant privacy, data integrity, or interoperability across diverse infrastructures.

6 Conclusion and Future Directions

The REMINDER framework showcases the potential of FL to address pressing challenges in both healthcare and smart buildings. In healthcare contexts such as atrial fibrillation detection and aortic coarctation diagnosis, FL enables hospitals to collaborate on model training without exposing sensitive patient data or violating stringent privacy regulations. By integrating secure aggregation methods, robust detection of adversarial updates, and privacy-preserving techniques like DP and HE, REMINDER empowers clinicians and researchers to leverage diverse medical datasets while maintaining compliance and safeguarding patient confidentiality.

In the case of smart buildings, the framework addresses efficient energy management by training predictive models on decentralized building data, preserving occupant privacy. This approach is particularly critical when fine-grained energy consumption patterns could reveal sensitive details about occupants' daily routines or building-specific operations. By equipping building management systems with DP and HE capabilities, robust aggregation schemes, and malicious client detection, REMINDER delivers real-time and reliable energy optimization. Furthermore, its interoperability layer and scalability provisions enable seamless integration of legacy building infrastructures alongside modern IoT devices, ensuring broad participation and collaborative innovation.

Across both healthcare and smart buildings, the REMINDER framework demonstrates that secure, privacy-focused FL can drive advanced data analytics without compromising sensitive information. By uniting robust security measures, adaptive learning mechanisms, and regulatory compliance features, this architecture paves the way for creating safer, more efficient, and better-informed environments—ultimately facilitating improved clinical outcomes and sustainable energy practices. As these use cases continue to evolve, REMINDER's modular and extensible design will remain key to meeting future demands for both data-driven healthcare solutions and intelligent building systems.

References

[1] G. Xia, J. Chen, C. Yu, and J. Ma, "Poisoning attacks in federated learning: A survey," *IEEE Access*, vol. 11, pp. 10708–10722, 2023.

[2] H. N. C. Neto, J. Hribar, I. Dusparic, D. M. F. Mattos, and N. C. Fernandes, "A survey on securing federated learning: Analysis of applications, attacks, challenges, and trends," *IEEE Access*, vol. 11, pp. 41928–41953, 2023.

[3] Z. Tian, L. Cui, J. Liang, and S. Yu, "A comprehensive survey on poisoning attacks and countermeasures in machine learning," *ACM Computing Surveys*, vol. 55, no. 8, pp. 1–35, 2022.

[4] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," *arXiv preprint arXiv:2003.02133*, 2020.

[5] P. Kiourti, K. Wardega, S. Jha, and W. Li, "Trojdrl: evaluation of backdoor attacks on deep reinforcement learning," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*, pp. 1–6, IEEE, 2020.

[6] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, "Analyzing federated learning through an adversarial lens," in *International Conference on Machine Learning*, pp. 634–643, PMLR, 2019.

[7] V. Shejwalkar and A. Houmansadr, "Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning," in *NDSS*, 2021.

[8] G. Baruch, M. Baruch, and Y. Goldberg, "A little is enough: Circumventing defenses for distributed learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[9] M. Fang, X. Cao, J. Jia, and N. Z. Gong, "Local model poisoning attacks to byzantine-robust federated learning," in *Proceedings of the 29th USENIX Conference on Security Symposium*, pp. 1623–1640, 2020.

[10] K. Nishimoto, Y.-H. Chiang, H. Lin, and Y. Ji, "Fedatm: Adaptive trimmed mean based federated learning against model poisoning attacks," in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, pp. 1–5, 2023.

[11] F. Colosimo and F. De Rango, "Median-krum: A joint distance-statistical based byzantine-robust algorithm in federated learning," in *Proceedings of the Int'l ACM Symposium on Mobility Management and Wireless Access*, MobiWac '23, (New York, NY, USA), p. 61–68, Association for Computing Machinery, 2023.

[12] E. M. Campos, A. Gonzalez-Vidal, J. L. Hernández-Ramos, and A. Skarmeta, "Fedrdf: A robust and dynamic aggregation function against poisoning attacks in federated learning," *IEEE Transactions on Emerging Topics in Computing*, 2024.

[13] W. Diffie and M. E. Hellman, "Multiuser cryptographic techniques," in *Proceedings of the June 7-10, 1976, National Computer Conference and Exposition*, AFIPS '76, (New York, NY, USA), p. 109–112, Association for Computing Machinery, 1976.

[14] A. E. Adeniyi, R. G. Jimoh, and J. B. Awotunde, "A systematic review on elliptic curve cryptography algorithm for internet of things: Categorization, application areas, and security," *Computers and Electrical Engineering*, vol. 118, p. 109330, 2024.

[15] J. Song and D. Namiot, "A survey of the implementations of model inversion attacks," in

International Conference on Distributed Computer and Communication Networks, pp. 3–16, Springer, 2022.

[16] X. Zhao, W. Zhang, X. Xiao, and B. Lim, "Exploiting explanations for model inversion attacks," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 682–692, 2021.

[17] J. Niu, P. Liu, X. Zhu, K. Shen, Y. Wang, H. Chi, Y. Shen, X. Jiang, J. Ma, and Y. Zhang, "A survey on membership inference attacks and defenses in machine learning," *Journal of Information and Intelligence*, 2024.

[18] M. Song, Z. Wang, Z. Zhang, Y. Song, Q. Wang, J. Ren, and H. Qi, "Analyzing user-level privacy attack against federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 10, pp. 2430–2444, 2020.

[19] H. Liu, B. Li, C. Gao, P. Xie, and C. Zhao, "Privacy-encoded federated learning against gradient-based data reconstruction attacks," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 5860–5875, 2023.

[20] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, and H. Vincent Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454–3469, 2020.

[21] P. Ruzafa-Alcázar, P. Fernández-Saura, E. Mármol-Campos, A. González-Vidal, J. L. Hernández-Ramos, J. Bernal-Bernabe, and A. F. Skarmeta, "Intrusion detection based on privacy-preserving federated learning for the industrial iot," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1145–1154, 2023.

[22] O. Choudhury, A. Gkoulalas-Divanis, T. Salonidis, I. Sylla, Y. Park, G. Hsu, and A. Das, "Anonymizing data for privacy-preserving federated learning," 2020.

[23] R. L. Rivest and M. L. Dertouzos, "On data banks and privacy homomorphisms," 1978.

[24] C. Gentry, "Fully homomorphic encryption using ideal lattices," vol. 9, pp. 169–178, 05 2009.

[25] V. Sidorov, E. Y. F. Wei, and W. K. Ng, "Comprehensive performance analysis of homomorphic cryptosystems for practical data processing," 2022.

[26] D. Bachlechner, R. Hetfleisch, S. Krenn, T. Lorünser, and M. Rader, "Protecting privacy in federated time series analysis: A pragmatic technology review for application developers," 08 2024.

[27] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," vol. 5, pp. 223–238, 05 1999.

[28] C. Zhou and N. Ansari, "Securing federated learning enabled nwdaf architecture with partial homomorphic encryption," *IEEE Networking Letters*, vol. 5, no. 4, pp. 299–303, 2023.

[29] E. M. Campos, A. G. Vidal, J. L. H. Ramos, and A. Skarmeta, "Fedrdf: A robust and dynamic aggregation function against poisoning attacks in federated learning," 2024.

[30] W. H. Organization, "Global health estimates 2020: Deaths by cause, age, sex, by country and by region, 2000-2019," tech. rep., 2020.

[31] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 10 2016.

[32] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, 2017.

[33] Z. Wang, Y. Huang, S. Mengkai, L. Wu, F. Xue, and K. Ren, "Poisoning-assisted property inference attack against federated learning," *IEEE Transactions on Dependable and Secure Computing*, vol. PP, pp. 1–13, 01 2022.

[34] C. Chen, L. Lyu, H. Yu, and G. Chen, "Practical attribute reconstruction attack against federated learning," *IEEE Transactions on Big Data*, vol. 10, no. 6, pp. 851–863, 2024.

[35] A. González-Vidal, A. P. Ramallo-González, and A. Skarmeta, "Empirical study of massive set-point behavioral data: Towards a cloud-based artificial intelligence that democratizes thermostats," in *2018 IEEE International Conference on Smart Computing (SMARTCOMP)*, pp. 211–218, 2018.

[36] E. M. Campos, A. G. Vidal, J. L. Hernández Ramos, and A. Skarmeta, "Federated transfer learning for energy efficiency in smart buildings," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1–6, 2023.

[37] A. Ntafalias, S. Tsakanikas, S. Skarvelis-Kazakos, P. Papadopoulos, A. F. Skarmeta-Gómez, A. González-Vidal, V. Tomat, A. P. Ramallo-González, R. Marin-Perez, and M. C. Vlachou, "Design and implementation of an interoperable architecture for integrating building legacy systems into scalable energy management systems," *Smart Cities*, vol. 5, no. 4, pp. 1421–1440, 2022.