IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Digital Object Identifier

ClusFed: A Clustering-Based Defense for Secure Federated Learning

HIBATALLAH KABBAJ[1,2], RACHID EL-AZOUZI[1], ABDELLATIF KOBBANE [2].
[1]LIA, CERI, University of Avignon, 84911, Avignon, France (e-mail: rachid.elazouzi@univ-avignon.fr)
[2]ENSIAS, Mohammed V University in Rabat, 10056, Morocco (e-mail: abdellatif.kobbane@ensias.um5.ac.ma)

Corresponding author: (e-mail: hibat-allah.kabbaj@alumni.univ-avignon.fr).

This work is supported by L'Agence nationale de la recherche (ANR) DELIGHT-22-CE23-0024 and the Alkhawarizmi Artificial Intelligence Project (grant number: Alkhawarizmi/2020/34).

ABSTRACT

As IoT devices continue to proliferate, the demand for secure and efficient machine learning solutions becomes increasingly critical. Federated Learning (FL) offers a promising approach by enabling decentralized model training across distributed clients while preserving data privacy. However, FL systems are susceptible to adversarial threats, such as data and model poisoning attacks, where compromised clients send corrupted updates to undermine the global model's performance. In this paper, we develop an innovative, lightweight Byzantine resistance strategy, called ClusFed, which uses adaptive client selection mechanisms. By leveraging clustering techniques to dynamically differentiate between honest and malicious clients, based on their local model updates. ClusFed effectively mitigates the impact of malicious nodes while sometimes exploiting their data, depending on the type of attack. This approach ensures robust performance even with up to 40% of clients being malicious. Extensive experiments on non independent, identically distributed (non-IID) partitions of MNIST, CIFAR-10, Shakespeare and a time series Water Leak datasets demonstrate that ClusFed consistently outperforms state-of-the-art Byzantine-resilient methods like FedMedian, Multi-Krum, and TrimmedMean. The results highlight ClusFed's ability to maintain high accuracy, achieving up to 93% for MNIST and 76% for CIFAR-10, 86% for Water Leak and 40% for Shakespeare dataset, as well as stability in various attack scenarios while guaranteeing efficient learning times.

INDEX TERMS Federated Learning, Adversarial Attacks, Cyber Defense, Distributed AI, Decentralized Machine Learning, Cybersecurity in IoT, Edge Computing.

I. INTRODUCTION

With the advancements in Internet of Things (IoT) technologies, the demand for secure and efficient machine learning solutions has become increasingly critical. IoT devices generate vast amounts of data, offering unprecedented opportunities for intelligent applications in sectors such as healthcare, finance, industy 4.0 and smart cities [1]–[4]. Recent advances in IoT resource management and game-theoretic optimization have also been demonstrated in related domains [5], [6]. However, these opportunities come with significant challenges, particularly regarding data privacy, security and scalability.

FL has emerged as a transformative approach in distributed machine learning, enabling decentralized model training across multiple devices while preserving data privacy [7], [8]. By allowing clients to train models locally on their data and only share model updates, FL minimizes the risk of data breaches. This paradigm has found applications in privacy-sensitive domains such as healthcare, finance, and IoT networks [9], [10], where sharing raw data is often infeasible. Google have leveraged FL for applications such as keyboard prediction while maintaining user privacy [11]. However, despite its advantages, FL is inherently vulnerable to adversarial threats due to its decentralized nature and its reliance on various types of clients, including potentially untrusted devices. These vulnerabilities are particularly pronounced in IoT environments, where devices may have limited computational resources and are more susceptible to compromise [12], [13].

Adversarial threats in FL typically manifest as two primary attack vectors: data poisoning and model poisoning. In data poisoning attacks, malicious clients inject manipulated or mislabeled data during local training to bias the global model's behavior [14], [15]. These attacks can lead to

degraded model performance or targeted misclassifications, posing significant risks in safety-critical applications such as autonomous driving or medical diagnosis [16]–[18]. On the other hand, model poisoning attacks involve adversaries directly modifying local model updates before sending them to the server, effectively bypassing the need for malicious training data. This type of attack can stealthily degrade global model accuracy or embed backdoors into the model [19] [20]. The decentralized nature of FL exacerbates these challenges by limiting the server's ability to verify client updates or monitor training processes [21], [22].

Furthermore, Byzantine attacks represent a broader class of adversarial behaviors where compromised clients send arbitrary or carefully crafted updates to disrupt the aggregation process [23] [24]. These attacks are particularly challenging because they exploit the lack of direct oversight in FL systems and can adapt dynamically to bypass static defenses [25]. For instance, well-crafted Byzantine attacks can mimic honest client behavior while subtly degrading global model performance over time [13]. The impact of such attacks is further amplified in non-IID settings, where client data distributions vary significantly, making it harder for traditional defenses to distinguish between legitimate and malicious updates [26] [27].

To mitigate these security challenges in FL, researchers have proposed several strategies aimed at improving system robustness. Among these, robust aggregation methods, such as trimmed-mean and Krum, have been widely explored for their ability to reduce the impact of malicious updates by employing statistical techniques designed to tolerate adversarial behavior [23] [28]. These methods work by filtering out extreme values that could indicate adversarial manipulation, thereby maintaining the integrity of the global model. However, these approaches come with inherent limitations that reduce their practicality in real-world FL environments.

One significant limitation is the computational overhead introduced by these methods. Their reliance on complex statistical computations requires additional resources and time to process client updates. This can result in training times that are up to 50% longer compared to traditional FL methods [25], making them less suitable for large-scale systems with limited computational capacity or strict real-time requirements. Furthermore, these techniques often assume that malicious clients are a minority and that honest clients' updates are statistically similar, which may not hold true in non-IID settings [26]. This assumption can lead to overfiltering, where legitimate updates from clients with unique data distributions are mistakenly discarded [27].

Moreover, robust aggregation techniques often struggle with balancing the need to filter out malicious updates while retaining valuable contributions from honest clients. In some cases, these methods may overfilter, removing not only adversarial updates but also legitimate contributions from clients whose data distributions differ from the majority. This overfiltering can negatively affect model performance in non-IID settings. Conversely, these techniques may also fail to detect well-crafted malicious updates, allowing adversarial clients to bypass defenses and gradually degrade global model accuracy over time [13].

In addition to aggregation-based defenses, other strategies such as anomaly detection and reputation systems have been proposed. Anomaly detection methods aim to identify suspicious client behavior by monitoring deviations from expected update patterns [21], [29]. While promising, these methods require extensive historical data and can be circumvented by sophisticated adversaries who mimic normal client behavior [13]. Reputation systems attempt to evaluate the trustworthiness of clients based on their historical contributions [30]. However, they often misclassify honest yet atypical update patterns as untrustworthy and still fail to detect well-crafted malicious participants, which limits their practical effectiveness.

On the other hand, threshold-based methods attempt to identify and exclude malicious clients using predefined criteria such as reputation metrics or contribution assessments [21], [25]. However, static thresholds are vulnerable to adaptive attacks that can lead to significant drops in model accuracy over time [21]. Clustering-based techniques have also been explored as a means of segregating malicious clients from honest ones through unsupervised learning [31]. While effective in identifying threats, these methods often demand substantial computational resources and rely on assumptions about client data distribution that may not hold true in practical scenarios [27].

The application of existing strategies in real-world FL settings presents several challenges. Many current methods, while prioritizing security, often fall short in adapting to the dynamic nature of adversarial attacks. These approaches frequently rely on static defenses that are ill-suited for evolving threats, leading to vulnerabilities when faced with sophisticated adversaries who can adapt their strategies over time [27]. Additionally, threshold-based and clustering-based methods often make unrealistic assumptions about data distribution and client behavior, which can result in significant performance declines when these assumptions are violated [21]. Such methods may struggle to effectively distinguish between malicious and honest clients in non-IID environments, where data heterogeneity is prevalent [26]. However, these strategies may fail to account for the subtlety of well-designed attacks that mimic legitimate client updates, allowing adversaries to bypass defenses and gradually degrade model performance [13], [32]. This highlights the need for innovative solutions capable of dynamically adapting to evolving attack patterns and providing robust protection without relying on rigid assumptions or static thresholds.

In this paper, we propose ClusFed, an new approach that balances security and efficiency by leveraging adaptive client selection mechanisms. Our strategy enhances the resilience of FL systems against Byzantine attacks while maintaining practical feasibility for deployment across diverse environments. By dynamically adjusting client selection based on clustering techniques that analyze local and global model

distances, ClusFed ensures robust performance even under challenging attack scenarios.

Our main contributions are:

- We propose ClusFed, a new client selection mechanism that uses adaptive threshold strategies to distinguish honest clients from malicious participants, overcoming the weaknesses of static threshold.
- Extensive experiments on the MNIST, CIFAR-10, Shakespeare and Water Leak datasets with non-IID partitions demonstrate superior accuracy and robustness against various types of adversarial attacks that we proposed, compared to existing methods such as FedAvg, Multi-Krum, and FedMedian.

We validated our approach on four different datasets; MNIST, CIFAR-10, Shakespeare, and a water-leak dataset, covering image, text, and time-series domains. Experimental results show that ClusFed consistently outperforms both vanilla FedAvg and leading robust aggregators, achieving up to 8–10% higher test accuracy under 40% malicious clients in diverse attack scenarios, and maintaining stable convergence across all non-IID partitions.

The remainder of this paper is organized as follows: Section II reviews related work on FL and the security strategies developed to address adversarial threats. Section III provides a comprehensive background on FL, including its workflow and a detailed problem formulation. Section IV introduces the proposed approaches including ClusFed and explaining its methodology. Section V describes the experimental setup and datasets, the proposed adversarial attack scenarios, the results obtained from ClusFed, evaluating its performance in comparison to existing client selection techniques. Section VI discusses open challenges and outlines potential future research directions in FL security. Finally, Section VII concludes the paper by summarizing the key contributions and emphasizing the significance of the findings.

II. RELATED WORK

In this section, we review the most relevant work on attacks such as Byzantine attacks and backdoor attacks, and defenses in FL. Backdoor attacks in FL [33] [34] aim to manipulate the global model so that it predicts a specific target class for inputs embedded with a backdoor trigger. These attacks are carried out by compromised clients that inject malicious updates during training. A Byzantine attack occurs when malicious clients intentionally send corrupted model updates to the server with the goal of disrupting the training process or degrading the global model's performance. For a more comprehensive review of backdoor attacks and corresponding defenses, we refer to [35].

Besides, data-poisoning attacks such as label-flipping, where clients randomly or systematically corrupt their labels, were shown by Tolpegin et al. to degrade non-IID MNIST performance even at low poison rates [15]. Model-poisoning by sign-flipping, simply reversing the sign of the honest update vector, was introduced by Li et al. and shown to evade many robust aggregators while severely harming convergence [36]. More recent work [37] studied minimal-trigger backdoors in FL, showing that even a single-character pattern in text data can be learned as a hidden functionality without hurting main-task accuracy.

There are two dominant defensive measures developed in the literature against such attacks, which can be classified into two categories: Robust aggregation [38] and Anomaly detection [29], [39], [40]. Robust Aggregation techniques focus on combining local patterns in a way that minimizes the impact of malicious attacks, whereas anomaly Detection aims to identify and eliminate malicious clients or corrupted local data. While defenses based on robust aggregation can be relatively effective against malicious nodes, their robustness is often limited to scenarios with only a few malicious clients [41]. Additionally, many robust aggregation methods rely on the availability of a clean, representative validation dataset on the server, which may not always be feasible. Fu et al. [42] developed Attack-Resistant Federated Averaging (ARFED), employing outlier elimination to enhance resilience against various attacks. Similarly, Pillutla et al. [43] introduced RFA, a robust aggregation oracle based on the geometric median. As the number of malicious clients increases, a more robust solution is required to effectively detect and mitigate malicious clients, thereby enhancing the robustness of FL in extreme scenarios [44]. In anomaly detection, several filtering methods have gained significant popularity, one of which is the Multi-Krum algorithm [23]. This method aggregates only the model updates that are closest to the barycenter of updates from all clients, effectively eliminating malicious contributions that are assumed to be outliers. A hybrid approach has been also developed by combining noise adding (robust training) with model clustering (filtering) to achieve state-of-the-art defense against FL backdoors [45]. Recent surveys, such as the one by Pian et al. [46], highlight the growing importance of integrating security and privacy considerations into core FL processes. Kim et al. [47] provided a comprehensive review of optimization techniques for client selection in FL, highlighting strategies to enhance learning efficiency and model performance while mitigating security risks.

Beyond robust aggregation and anomaly detection, a parallel line of work in FL focuses on privacy-preserving protocols. Bonawitz et al. proposed a Secure Aggregation protocol that enables the server to aggregate client updates without ever seeing individual model weights, protecting against honest-but-curious servers and colluding clients [48]. Geyer et al. proposed a client-level differential-privacy mechanism that adds calibrated noise during each round to protect individual clients' updates while training in federated settings [49], while Truex et al. developed a per-client local DP framework that balances privacy and model utility in highly heterogeneous settings [50]. These techniques ensure that even aggregate statistics leak minimal information about any single client's data. Another important dimension is client-selection and scheduling, which aims to improve both efficiency and resilience by choosing subsets of clients each round. Nishio formulated the resource-aware FedCS protocol

TABLE 1: Comparative overview of federated-learning defenses.

Strategies	Defense type	Attack types	Non-IID?	Data type	Adversary %	Complexity
[7]	Baseline averaging (no explicit Byzantine defense)	—	Yes	image/text/language	N/A	$\mathcal{O}(N \cdot d)$
[23]	Selection / nearest-to-barycenter selection of updates	Byzantine/model poisoning	Limited, degrades under strong heterogeneity	synthetic/image	Theoretical tolerance: $< 50\%$; practical tests $\leq$ 20–40%	$\mathcal{O}(N^2 \cdot d)$
[28] [53]	Coordinate-wise robust aggregation (trim/median)	Byzantine/model poisoning	Theory often assumes iid/sub-Gaussian; degraded under strong non-IID	synthetic/image	Per-coordinate breakdown up to $\approx 50\%$ (practically $\leq 40\%$)	$\mathcal{O}(d \cdot N \log N)$
[43] [42]	Geometric-median / residual reweighting / outlier elimination	data/model poisoning/backdoor variants	Yes	image/text	Empirical testing typically $\lesssim$ 40%; theoretical breakdown $\approx 50\%$	$\mathcal{O}(N \cdot d)$
[31] [27] [45]	Clustering of updates, adaptive thresholding; hybrid (clustering+noise/clipping)	Byzantine, backdoor, outliers	Yes, explicitly designed/evaluated for non-IID	image/text/time-series	Empirical robustness reported up to $\approx$ 40–50% in select settings	$\mathcal{O}(N \cdot d + N \log N)$
[54] [53]	Distance scoring / outlier detection based on model distances	data/model poisoning	Moderate, can overfilter under strong non-IID	image/time-series	Empirical: moderate robustness $\approx$ 30–40% (dataset dependent)	$\mathcal{O}(N \cdot d + N \log N)$
[48] [49], [50]	Cryptographic secure aggregation; differential privacy (noise addition)	Privacy threats/honest-but-curious	Yes, data distribution agnostic	image/text/time-series	N/A (not a Byzantine defence)	$\mathcal{O}(N \cdot d)$
ClusFed	**Clustering-based client selection + robust aggregation**	**data/model poisoning**	**Designed for non-IID**	**image/text/time-series**	**Evaluated at 40% malicious (robust in experiments)**	$\mathcal{O}(N)$

to prioritize clients with high bandwidth and computational capacity, reducing training latency under mobile edge constraints [51]. Wang et al. later derived optimal selection policies that maximize convergence speed under non-IID data distributions by solving a knapsack-style optimization over client heterogeneity [52]. Clustering-based defenses have emerged as a flexible way to both detect and exclude malicious updates without relying on strong assumptions about the number or behaviour of compromised clients. Sattler et al. introduced a Robust Federated Aggregation (RFA) scheme that uses a mixture of Gaussians to cluster model updates, discarding those that fall in small, outlier clusters [31]. Shen et al. extended this idea with hierarchical clustering and adaptive thresholding, demonstrating strong robustness even when up to 50% of the clients are Byzantine [27].

Table 1 summarizes key properties and trade-offs of representative defensive approaches discussed above.

Unlike filtering-based defenses [23], [38], ClusFed dynamically and adaptively groups client updates based on their mutual similarity and selects only the most coherent subgroup for aggregation, allowing it to tolerate a much higher fraction of adversaries without excluding legitimate but diverse honest contributions.

III. PROBLEM DESCRIPTION

We begin by reviewing the basic FL framework and the standard Federated Averaging (FedAvg) algorithm, initially without considering any malicious attacks. Subsequently, we define our system model, incorporating the presence of malicious participants, and discuss the resilience of the learning process to corrupted model updates in this adversarial setting.

TABLE 2: Table of Notations

Symbol	Description
ω_t	Global model parameters at round t.
ω_t^k	Local model parameters of client k at round t.
$\mathcal{S}_t$	Set of selected clients at round t.
$\mathcal{K}$	Total set of clients in the FL system.
$F_k(\omega)$	Local objective function for client k.
η	Learning rate used during local training.
p_k	Aggregation weight for client k, typically $\frac{n_k}{\sum_{i \in \mathcal{S}_t} n_i}$.
n_k	Number of data points at client k.
M_k^t	Euclidean distance between ω_t^k and ω_t, i.e., $\|\omega_t^k - \omega_t\|_2$.
λ	Penalty term for adding an additional cluster.
σ^2	Global variance of distances in ClusFed.
μ_0	Mean of distances $\overline{M}_c$, i.e., $\mu_0 = \frac{1}{N}\sum_{c=1}^{N} \overline{M}_c$.
τ^t	Adaptive threshold for distance-based weighting, $\tau^t = \mu_M + \alpha \cdot \sigma_M$.
ϵ	Small constant to avoid division by zero.
$\mathcal{S}_t^h$	Set of honest clients at round t.
$\mathcal{S}_t^b$	Set of Byzantine (malicious) clients at round t.
j^*	Optimal split point minimizing the total cost of grouping clients into one or two clusters in ClusFed.
$C(i,j)$	Cost of grouping $\{\overline{M}_{\gamma(i)}, \ldots, \overline{M}_{\gamma(j)}\}$ into one cluster with the optimal choice of centroid.

A. FEDERATED LEARNING REVIEW

The decentralized approach of FL is particularly advantageous in privacy-sensitive domains such as healthcare, finance, and IoT applications. This paradigm not only enhances data privacy but also facilitates the training of more robust models by leveraging diverse data sources from various clients.

FL operates through a systematic workflow that involves iterative steps, enabling collaboration among distributed

clients while maintaining data privacy. These steps form the foundation of the FL paradigm, ensuring effective model training across diverse and decentralized datasets. The FL process can be described as follows:

1. Initialization: The server initializes the global model parameters ω_0.
2. Client Selection: In each round t, the server selects a subset of clients $\mathcal{S}_t$ from the total set of clients $\mathcal{K}$.
3. Local Training: Each selected client $k \in \mathcal{S}_t$ updates the local model ω_t^k using its private data $\mathcal{D}_k$:

$$\omega_t^k = \omega_{t-1} - \eta \nabla F_k(\omega_{t-1}) \tag{1}$$

where η is the learning rate and F_k is the local objective function.
4. Model Upload: Clients send their updated models ω_t^k to the server.
5. Aggregation: The server aggregates the received models to update the global model:

$$\omega_t = \sum_{k \in \mathcal{S}_t} p_k \omega_t^k \tag{2}$$

where p_k are aggregation weights, typically set to $\frac{n_k}{\sum_{i \in \mathcal{S}_t} n_i}$, with n_k being the number of data points at client k.
6. Model Update: The server updates the global model, and the process repeats from step 2 until convergence or a predefined number of rounds.

B. PROBLEM FORMULATION

The integrity of the FL process is threatened by malicious clients aiming to disrupt the global model's training. These adversarial influences can manifest through erroneous updates or manipulated local datasets, complicating model parameter aggregation. To counter these threats, it's essential to design strategies that detect and mitigate malicious client effects while ensuring robust and efficient learning.

In FL, Byzantine attacks involve malicious clients disrupting the learning process [23]. These attacks can take various forms, as shown in Figure 1:

Model Poisoning: Malicious clients send crafted model updates to bias the global model [13].
Data Poisoning: Adversaries manipulate local datasets to indirectly affect the global model [25].
Evasion Attacks: Malicious clients evade detection mechanisms while negatively influencing the model [33].

As illustrated in Figure 1, the FL workflow is vulnerable to adversarial influences at multiple stages of the training and prediction phases. During the local training phase, data poisoning attacks can occur when malicious clients introduce corrupted data into their local datasets, skewing indirectly the updates sent to the server. This manipulation compromises the integrity of the global model by embedding biases or inaccuracies. Similarly, during the model aggregation phase, model poisoning attacks can take place as adversarial clients send manipulated updates to the server with the intent to degrade directly the global model's performance. These attacks aim to either mislead the global model or embed backdoors for later exploitation. Finally, in the prediction phase, evasion attacks exploit vulnerabilities in the trained global model by crafting inputs that cause incorrect predictions, further undermining its reliability and utility.

The presence of Byzantine clients can significantly degrade FL system performance and reliability. Formally, we model this problem as:

$$\omega_t = \text{Aggregate}(\{\omega_t^k \mid k \in \mathcal{S}_t^h\} \cup \{\omega_t^m \mid m \in \mathcal{S}_t^b\}), \tag{3}$$

where $\mathcal{S}_t^h$ is the set of honest clients and $\mathcal{S}_t^b$ is the set of Byzantine clients, with $\mathcal{S}_t = \mathcal{S}_t^h \cup \mathcal{S}_t^b$.

The main challenge is to design a robust aggregation method that can:

1. Identify and mitigate the impact of Byzantine clients.
2. Maintain convergence and performance of the global model.
3. Operate effectively under non-IID data distributions typical in FL settings.

Evaluation Metrics

To assess our approach's effectiveness against Byzantine attacks in FL, we use server test accuracy as a primary evaluation metric. This metric is defined as the proportion of correctly classified samples in a held-out test dataset consisting of benign data points. By monitoring server test accuracy, we can quantify our model's performance in the presence of adversarial clients, ensuring the integrity and reliability of the FL process.

IV. CLUSFED

In this section, we delve into the server's approach for selecting clients whose local models will contribute to updating the global model. The selection process can leverage the metric M_c, which quantifies the distance or utility for each client c. One widely used approach involves probabilistic selection based on scores or utility values, employing the Gibbs or Boltzmann distribution $\frac{\exp(-u_i/\tau)}{\sum_{k \in \mathcal{S}} \exp(-u_k/\tau)}$, where τ is a positive parameter known as the temperature. Adjusting τ influences the selection process; increasing its value reduces the number of clients chosen in each round. While this method is effective for probabilistic client selection, it lacks the ability to differentiate between honest and malicious clients. Additionally, it often requires numerous rounds to achieve meaningful client selection, making it less suitable in adversarial scenarios.

A. ROBUST AGGREGATION: FROM DISTANCE-BASED FL (DISTFL) TO WEIGHTED MEDIAN AGGREGATION (WMA)

Robust aggregation strategies are critical in FL to ensure resilience against Byzantine attacks, which can severely degrade the performance of the global model. In this work, we incorporate elements from DistFL [54] to address these challenges effectively.

The DistFL approach introduces a distance-based metric to detect and isolate unreliable or malicious clients during training. In FL systems, malicious clients often send updates

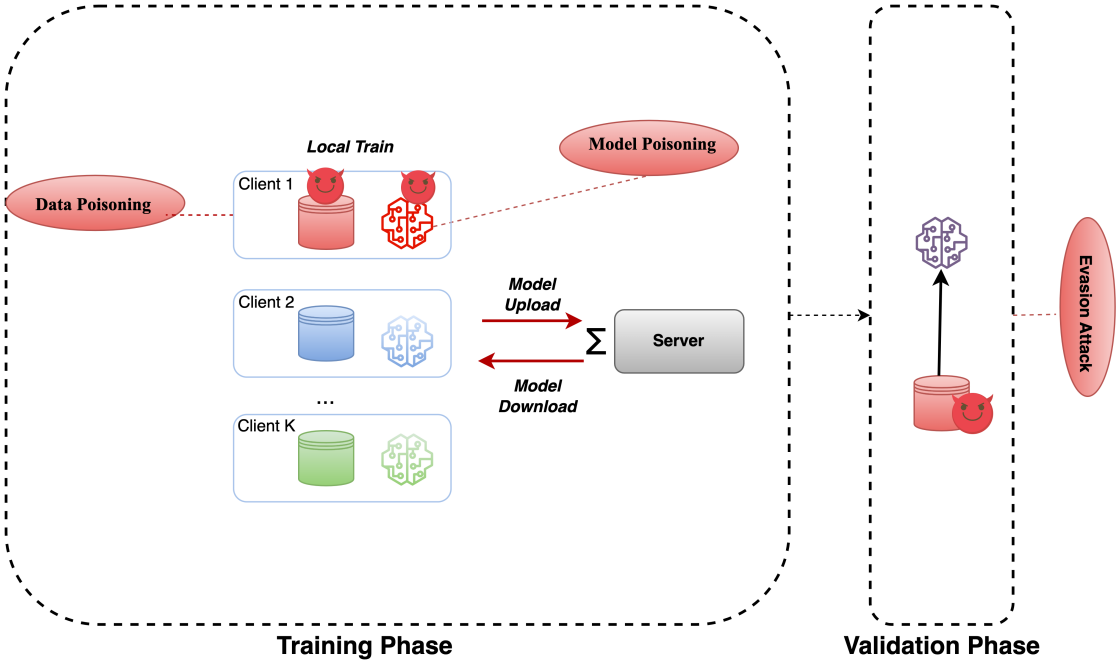


FIGURE 1: FL System with different Attack types.

that deviate significantly from those of honest clients due to adversarial intent or data heterogeneity. The Euclidean distance M_k^t measures the deviation between a client's local model parameters ω_t^k and the global model parameters ω_t. By quantifying these deviations using distance, DistFL provides a mechanism for identifying outliers. The distance for each client k at round t is defined as:

$$M_k^t = \|\omega_t^k - \omega_t\|_2, \tag{4}$$

where M_k^t represents the Euclidean distance for client k, ω_t^k is the local model of client k, and ω_t is the global model at round t. M_k^t is computed during each training round to evaluate client reliability dynamically.

Despite its strengths, DistFL has several limitations that constrain its applicability in FL environments. While it effectively identifies and isolates malicious clients by leveraging a distance-based metric, it focuses solely on client detection and filtering without addressing how to robustly aggregate updates from the remaining clients. This binary classification of clients as either honest or malicious, based on a threshold applied to the Euclidean distance M_k^t, can lead to overly strict exclusions.

As a result, honest clients with slightly divergent updates, potentially caused by their unique data distributions or statistical heterogeneity, may be unfairly categorized as malicious and excluded from all subsequent training rounds. In scenarios where only a subset of clients participates in training during each round, DistFL occasionally suffers from the unintended elimination of honest clients, as an honest client might be mistakenly classified as malicious due to low participation in training. This rigid exclusion deprives these clients of the opportunity to adapt or improve their contributions in future rounds, which could otherwise enrich the diversity and robustness of the global model. Furthermore, by completely removing these clients from the training process, DistFL risks reducing the overall pool of participating clients, which can negatively impact convergence and model generalization.

To address these challenges, we introduced Weighted Median Aggregation (WMA) [53], which builds upon DistFL's distance-based metric but incorporates a more nuanced approach to aggregation. Instead of relying solely on client detection and exclusion, WMA combines distance-based weighting with median-based aggregation to ensure robustness against outliers while preserving valuable contributions from honest clients.

The median serves as a robust central tendency measure that is inherently resistant to extreme values, making it well-suited for scenarios involving adversarial updates. For each model parameter f, WMA computes a weighted median, where weights are derived from the distance metric. Clients whose updates are closer to the global model are assigned higher weights, ensuring their greater influence in the aggregation process. This approach achieves three key objectives: i) It prioritizes reliable updates without rigidly excluding others. ii) It diminishes the impact of malicious clients by down-weighting their contributions. iii) It allows honest clients with slightly divergent updates due to non-IID data distributions to contribute meaningfully.

Formally, the weight assigned to each client k is defined as:

$$\text{weight}_k = \begin{cases} \frac{1}{M_k^t + \epsilon} & \text{if } M_k^t \leq \tau^t, \\ 0, & \text{otherwise,} \end{cases}$$

where $\epsilon > 0$ prevents division by zero, and $\tau^t = \mu_M + \alpha \cdot \sigma_M$ is an adaptive threshold based on the mean (μ_M) and standard deviation (σ_M) of distances across all participating clients.

By integrating these weighted contributions into a median-based framework, WMA ensures that unreliable updates are effectively down-weighted without compromising convergence. This balance between robustness and adaptability makes WMA particularly effective in adversarial settings.

However, while Weighted Median Aggregation (WMA) improves upon traditional aggregation methods by mitigating the influence of Byzantine clients through distance-based weighting and median-based aggregation, it still has limitations. WMA relies heavily on adaptive thresholds and static weighting schemes, which may fail to capture the dynamic nature of client behaviors in FL.

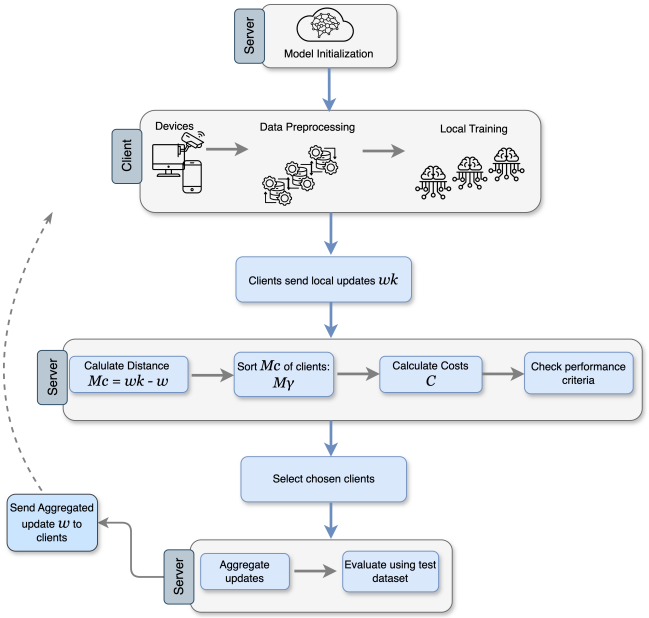


FIGURE 2: ClusFed Process: Client Selection, Distance Calculation, and Aggregation for Robust FL.

B. CLIENT SELECTION: CLUSTERING-BASED FEDERATED LEARNING (CLUSFED)

To address the aforementioned challenges, we propose ClusFed, a novel clustering-based client selection strategy that builds upon the distance metric M_k^t but introduces a more sophisticated mechanism for separating honest and malicious clients. By leveraging clustering techniques, ClusFed not only detects malicious clients but also enhances the aggregation process by dynamically adapting to evolving adversarial patterns. The core idea behind ClusFed is to group clients into clusters based on the distance between their local model updates and the global model. Unlike WMA, which assigns weights to all clients based on their M_k^t, ClusFed explicitly separates clients into distinct clusters, enabling more precise identification of malicious participants. This approach ensures that honest clients with slightly divergent updates due to non-IID data distributions are not unfairly penalized, while malicious clients are effectively isolated.

In ClusFed, the metric M_k^t serves as the foundation for clustering, enabling the separation of clients into two groups: one representing honest clients and the other representing malicious ones. The objective of this clustering process is to minimize the overall variance within each group while ensuring that the groups are distinct. Formally, given the set of distances $\overline{\mathbf{M}} = (\overline{M}_1, \overline{M}_2, \cdots, \overline{M}_N)$, the goal is to find centroids μ_1^* and μ_2^* that minimize the sum of squared deviations of distances from their respective centroids. This can be expressed as:

$$(\mu_1^*, \mu_2^*) = \arg\min_{\mu_1,\mu_2} \sum_{k\in\mathcal{S}} \min_{\mu\in\{\mu_1,\mu_2\}} (\overline{M}_k - \mu)^2. \quad (5)$$

The optimization problem in ClusFed involves determining how to partition the set of clients into two groups based on their distance metrics, minimizing the overall cost of grouping while ensuring the groups are distinct. To achieve this, the distances are arranged in ascending order, denoted as $\overline{M}_{\gamma(1)} \leq \overline{M}_{\gamma(2)} \leq \cdots \leq \overline{M}_{\gamma(N)}$, where $\gamma(i)$ represents the client at position i in the ordered list. For any subset of clients indexed from i to j, the cost of grouping these clients into a single group is calculated as:

$$C(i,j) = \sum_{k=i}^{j} (\overline{M}_{\gamma(k)} - \mu)^2,$$

where the centroid of this group is given by:

$$\mu = \frac{1}{j+1-i} \sum_{k=i}^{j} \overline{M}_{\gamma(k)}.$$

The process identifies the optimal split point j^* by finding the index that minimizes the total cost of forming two groups, expressed as:

$$j^* = \arg\min_j [C(1,j) + C(j+1,N)].$$

Thus, the solution of problem (5) is given by

$$\mu_1^* = \frac{1}{j^*+1-i} \sum_{k=i}^{j^*} \overline{M}_{\gamma(k)}, \quad (6)$$

and

$$\mu_2^* = \frac{1}{N-j^*} \sum_{k=j^*+1}^{N} \overline{M}_{\gamma(k)}. \quad (7)$$

Here, μ_1^* (resp. μ_2^*) represents the centroid of the first cluster, typically corresponding to honest (resp. malicious) clients, indexed from from i to j^* (resp. j^*+1 to N).

This formulation ensures that clients with similar updates are grouped together, while those with significantly different updates are placed in separate groups. The decision to split into two groups or keep all clients in a single group depends on whether splitting sufficiently reduces the total cost compared to maintaining a single group. This dynamic clustering

mechanism allows ClusFed to adaptively distinguish between honest and malicious clients based on their distance metrics, ensuring effective client selection for aggregation.

To achieve this adaptive grouping, ClusFed builds apon the principles of clustering by employing a regularized version of k-Means clustering [55]. In classical k-Means, the objective is to minimize intra-cluster variance (i.e., the sum of squared distances between data points and their assigned cluster centroid), by assigning each data point to the cluster with the nearest centroid. Inspired by this approach, ClusFed adapts the clustering process to the FL setting by dynamically determining whether the set of clients should be partitioned into one or two clusters based on their behavior during training. This adaptation is achieved by incorporating a penalty term $\lambda > 0$ into the clustering objective, which balances the trade-off between minimizing intra-cluster variance and penalizing unnecessary cluster formation. Unlike traditional k-Means, which requires the number of clusters (k) to be predefined, ClusFed's regularized clustering framework allows for dynamic selection, enabling it to adaptively separate honest clients from malicious ones based on their metric M_k^t. This regularization ensures that clustering decisions are robust to varying client behaviors and adversarial patterns, providing a more flexible and effective solution for client selection in FL. The regularized optimization problem is formulated as follows:

$$\min\left(\min_{\mu}\sum_{k\in\mathcal{S}}(\overline{M}_k-\mu)^2,\ \min_{\mu_1,\mu_2}\sum_{k\in\mathcal{S}}\min_{\mu\in\{\mu_1,\mu_2\}}(\overline{M}_k-\mu)^2+\lambda\right). \tag{8}$$

Clearly, the optimal μ^* for one cluster is the centroid and by embedding the optimal solution of (5), the optimization problem (8) becomes

$$\min\left(N\sigma^2, C(1,j^*)+C(j^*+1,N)+\lambda\right). \tag{9}$$

Therefore, the optimal solution of (9) indicates that if

$$C(1,j^*)+C(j^*+1,N)\leq N\sigma^2-\lambda, \tag{10}$$

then splitting the clients into two groups is justified. The algorithm ClusFed splits the set of clients into two clusters, and only $\{\gamma(1),..,\gamma(j^*)\}$ clients are chosen for computing the global model. The term $N\sigma^2-\lambda$ therefore serves as a threshold for deciding which nodes will be excluded from model learning and considered malicious nodes, so that learning continues with the clients $\{\gamma(1),...,\gamma(j^*)\}$. Conversely, if inequality (10) is not satisfied, all clients are grouped into a single cluster, meaning that no clear separation exists between honest and malicious clients based on their distance metrics. The value of λ is used as a guarantee to achieve a desired level of separation between two groups when ClusFed decides to switch from one cluster to two clusters. However, the parameter λ plays a critical role in balancing robustness and sensitivity in ClusFed's clustering process:

- High λ (Conservative Strategy): A higher value of λ makes ClusFed more conservative by favoring single-cluster solutions unless there is strong evidence for separation. This approach reduces false positives (i.e., misclassifying honest clients as malicious), which is particularly important in IID settings where client updates are naturally similar.
- Low λ (Aggressive Strategy): A lower value of λ encourages splitting even when there is moderate evidence for separation. This approach increases sensitivity to subtle adversarial patterns but may lead to higher false positives in non-IID settings where honest clients may exhibit greater variability in their updates.

The choice of the λ value depends on several factors, such as data distribution and the behavior of malicious nodes. When data is IID between clients, a higher λ value ensures that minor variations do not trigger unnecessary splits. On the other hand, when data distribution across clients is non-IID, we need to design a lower λ to account for greater heterogeneity among honest clients. With regard to the behavior of malicious nodes, when malicious clients exhibit subtle deviations from honest ones (e.g., well-designed adversarial updates), a lower λ improves detection accuracy by enabling finer separation.

The ClusFed method requires very few calculations, as it only requires two operations: sorting the distance $\overline{M}_k$ of the N clients and comparing N values to calculate the index j^*. Our method results in a complexity of order $\mathcal{O}(n)$.

Algorithm 1 CLUSFED: Clustering-Based Client Selection Strategy

INPUTS: Set of clients $\mathcal{S}$, distance vector $\overline{M} = (\overline{M}_1, \overline{M}_2, \ldots, \overline{M}_N)$, penalty parameter λ, and round t.
OUTPUT: Selected client set $\mathcal{S}^t$ for aggregation.
Sort distances $\overline{M}_k$: $\overline{M}_{\gamma(1)} \leq \overline{M}_{\gamma(2)} \leq \cdots \leq \overline{M}_{\gamma(N)}$.
Compute global variance: $\sigma^2 = \frac{1}{N}\sum_{c=1}^{N}(\overline{M}_c - \mu_0)^2$.
for $j = 1, \ldots, N$ **do**
    Compute costs:
    $C(1, j-1)$, $C(j, N)$.
end for
Identify optimal split point: $j^* = \arg\min_j(C(1,j) + C(j+1,N))$.
if $C(1,j^*) + C(j^*+1,N) + \lambda < N\sigma^2$ **then**
    Assign clients to two clusters:
    $\mathcal{S}_{\text{h}}^t = \gamma(1), ..., \gamma(j^*)$,
    $\mathcal{S}_{\text{b}}^t = \gamma(j^*+1), ..., \gamma(N)$.
    Select honest clients for aggregation: $\mathcal{S}^t = \mathcal{S}_{\text{h}}^t$.
else
    Group all clients into a single cluster: $\mathcal{S}^t = \mathcal{S}$.
end if
return $\mathcal{S}^t$

V. EXPERIMENTS

Our experimental setup aimed to evaluate the effectiveness of the ClusFed strategy in mitigating Byzantine attacks in

FL environments. We conducted extensive simulations using the Flower framework (version 1.8.0) [56], Python 3.8, TensorFlow 2.4.0, and Keras 2.4.3. We leveraged the computational resources of our laboratory's GPU cluster, specifically utilizing a server equipped with an NVIDIA RTX 2080Ti GPU, which has 11GB of VRAM. The server specifications included a 20-core CPU and 192 GB of RAM.

A. SIMULATION ENVIRONMENT

We established a FL environment with 80 clients, each possessing a unique subset of the MNIST, CIFAR-10, Shakespeare, and water leak datasets.

The MNIST dataset, comprising 60,000 training images and 10,000 test images of handwritten digits, was partitioned among the clients using a Dirichlet partitioner [57] with an α parameter of 0.5. This approach created a non-IID data distribution, reflecting real-world scenarios where clients have access to different subsets of data.

The neural network architecture employed for MNIST was tailored to its characteristics: an input layer (28x28 neurons), a flattening layer, a dense layer with 128 neurons and ReLU activation, a dropout layer (rate=0.2) for regularization, and an output layer with 10 neurons and softmax activation. We used the Adam optimizer with a learning rate of 0.001 and sparse categorical cross-entropy loss function.

For CIFAR-10, we employed a convolutional neural network (CNN) with multiple convolutional layers, max-pooling layers, and dense layers suitable for processing the more complex color images in CIFAR-10. The dataset consists of 50,000 training images and 10,000 test images across 10 classes. Similar to MNIST, we used a Dirichlet partitioner to create non-IID data distributions among clients.

The CNN architecture included several convolutional layers followed by max-pooling layers to capture spatial hierarchies in the data, culminating in fully connected dense layers for classification. The model was optimized using the Adam optimizer with a learning rate of 0.001 and categorical cross-entropy loss function.

For Shakespeare, we employed the federated Shakespeare next-character prediction dataset, which contains approximately 4.3 million character windows extracted from Shakespeare's plays. We partitioned the data across 80 clients where each client holds sequences of dialogue from a single character. Each training example is a sliding window of length 80 characters, with the next character as the prediction target. We built a shared vocabulary by collecting all characters across partitions, resulting in around 90 unique tokens (including padding). Our sequence model consists of an embedding layer with 64-dimensional outputs, a single LSTM layer with 128 units, and a softmax output over the vocabulary. Local training on each client ran using an Adam optimizer (learning rate 0.001) and sparse categorical cross-entropy loss. For evaluation, we held out the last 10% of each client's windows to compute next-character top-1 accuracy, then aggregated these scores centrally.

The dataset we employed for our leak-size classification task [58], comprises 18 files, each containing 150 000 sequential sensor readings of pressure and flow rate recorded at 100 Hz along a 65 m pipe with artificial leaks introduced at three locations (15 m, 30 m, 50 m) and three diameters per location. To simulate realistic federated heterogeneity, we partitioned the training set among 80 clients using a Dirichlet distribution so that some clients saw predominantly large leaks while others saw mostly small or no leaks. Each client trained a fully connected network (128 and 64 neurons in two hidden layers, ReLU activations) with a softmax output over the leak-size classes, using the Adam optimizer (learning rate 0.001) and categorical cross-entropy loss for five local epochs (batch size 32) per round.

This setup allowed us to evaluate the ClusFed strategy's robustness and efficiency across different datasets and model architectures.

B. NON-IID DATA PARTITIONING

In FL, handling non-IID data distributions is a critical challenge due to the diverse and decentralized nature of data sources. Our approach leverages Dirichlet partitioning to simulate realistic non-IID scenarios across clients. This method effectively distributes the MNIST and CIFAR-10 datasets among 80 clients, each receiving data with varying label distributions.

The Dirichlet partitioner was configured with an α parameter of 0.5, which controls the degree of non-IID-ness by influencing how unevenly labels are distributed among clients. A smaller α results in more skewed distributions, reflecting real-world cases where certain clients may have access to limited or biased data subsets [59].

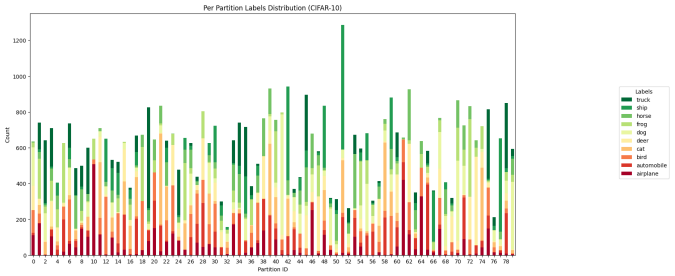
To ensure meaningful participation from all clients, a minimum partition size was set, preventing excessively small datasets that could hinder local model training. The self-balancing feature of the partitioner further aids in maintaining a reasonable distribution of data across clients.

Figure 3 illustrates the label distribution for MNIST and CIFAR-10 after Dirichlet partitioning, highlighting the non-IID nature of the dataset. This setup allows us to rigorously evaluate the robustness and adaptability of our ClusFed strategy under diverse data conditions.

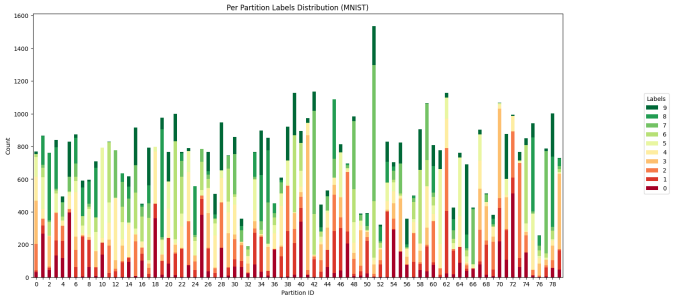
By employing this partitioning strategy, our experiments aim to mimic the heterogeneous data environments typical in FL deployments, providing insights into how well our approach can handle such variability while maintaining model performance.

C. ATTACK SIMULATION

In our experiments, we employed label-flipping attacks to evaluate the robustness of the ClusFed strategy. Label-flipping is a well-known data poisoning attack in FL, where malicious clients intentionally flip the labels of their local training data to disrupt the global model's performance [13], [60]. This attack is particularly relevant because it is easy to execute, difficult to detect, and can significantly degrade



(a) CIFAR-10



(b) MNIST

FIGURE 3: Label Distribution Across 80 Partitions (Non-IID Data).

model accuracy [15], [61]. Furthermore, label-flipping attacks are widely studied in the literature as a benchmark for evaluating the resilience of FL systems against adversarial threats [62], [63]. We chose label-flipping attacks for this study due to their versatility and applicability across different datasets and scenarios. These attacks simulate realistic adversarial behavior by targeting specific classes or applying global mislabeling patterns, making them ideal for testing the robustness of FL strategies. Additionally, label-flipping attacks are particularly challenging in non-IID settings, where client data distributions vary significantly, further stressing the system's ability to isolate malicious updates [60]. The attacks were applied to both CIFAR-10 and MNIST datasets, leveraging their distinct characteristics to analyze ClusFed's performance under varying data complexities. CIFAR-10, with its diverse classes and complex image features, presents a more challenging setting compared to MNIST's simpler grayscale digit images. By employing multiple variations of label-flipping attacks, including global and targeted mislabeling patterns, we aimed to comprehensively evaluate ClusFed's resilience in diverse adversarial scenarios.

We implemented four types of label-flipping attacks for MNIST and CIFAR-10:

Random Modification of Labels (RML): This attack introduces noise into the training process by systematically altering all labels in the dataset. A specific offset is added to each label, resulting in uniform corruption across all classes. This disrupts the natural associations between features and labels, significantly degrading the model's ability to generalize.

Random Modification of Labels for One Label Only (RML_1L): In this targeted attack, only a single class is affected. Labels corresponding to a specific target class are shifted by an offset, while all other labels remain unchanged. This focused corruption biases the model's predictions for the targeted class, leaving other classes unaffected but still disrupting overall performance.

Fixed Modification of Labels (FML): This attack applies a fixed offset to all labels throughout training. Unlike RML, where the offset may vary across rounds or clients, FML uses a consistent mislabeling pattern. This persistent corruption introduces systematic noise across all classes, making it particularly disruptive to model convergence over time.

Fixed Modification of Labels for One Label Only (FML_1L): Combining the targeted nature of RML_1L with the consistency of FML, this attack focuses exclusively on a single class while applying a fixed offset to its labels. The result is a persistent and targeted corruption that biases predictions for one class while leaving others stable. This focused yet consistent mislabeling can severely impact model accuracy for specific tasks.

As for Shakespeare and Water Leak datasets, we simulated a Sign-Flipping (SF) model-poisoning attack [64]. Malicious clients simply negate the entire model update before returning it to the server, simulating an extreme gradient manipulation attack without data poisoning. A subset of 32 out of 80 clients (40%) were designated as malicious.

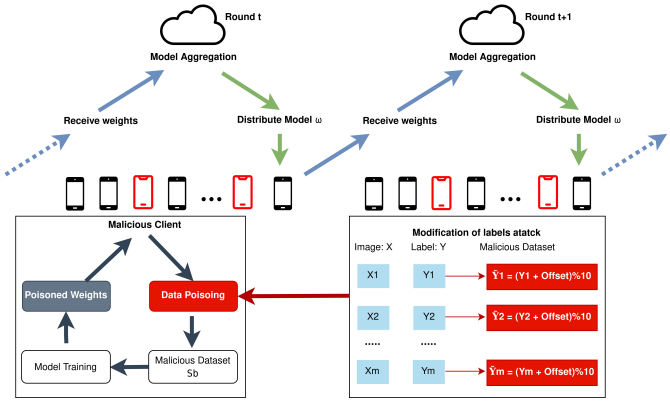


FIGURE 4: General Workflow of the Proposed Attacks

D. IMPLEMENTATION DETAILS

For MNIST and CIFAR-10, the attacks were applied during client-side training for malicious clients only(Figure 4). Each malicious client corrupted its local dataset using one of the above methods before sending updates to the server.

With CIFAR-10, which has more complex features and inter-class similarities, these attacks caused significant disruption by introducing semantically distant mislabeling (e.g., flipping "cat" to "truck"). For MNIST, with its simpler structure, all four attacks were effective in degrading model performance by biasing specific digits.

In all experiments, the penalty term λ in ClusFed's regularized clustering objective was set to $\frac{\sigma^2}{2}$, where σ^2 denotes

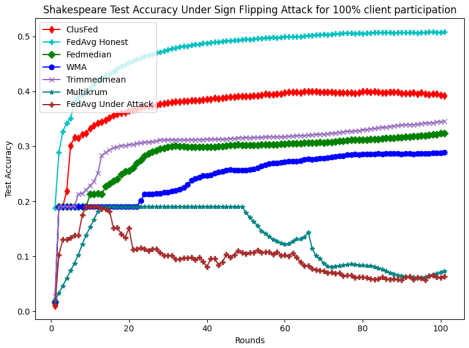


FIGURE 5: Test Accuracy over 100 Rounds for Different FL Strategies With Shakespeare Dataset under Sign Flipping attack (80 Clients, 40% Malicious)

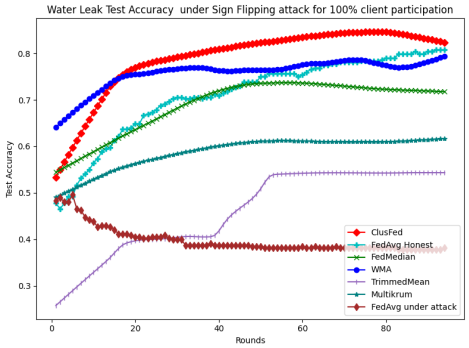


FIGURE 6: Test Accuracy over 100 Rounds for Different FL Strategies With Water leak Dataset under Sign Flipping attack (80 Clients, 40% Malicious)

the global variance of client distances (previously defined in Section III). This value was chosen as it provided the most balanced trade-off between separating honest and malicious clients while maintaining robust performance across different datasets and attack scenarios.

E. EVALUATION SETUP

To evaluate ClusFed's robustness under these attacks:

- We initiated attacks after an initial learning phase on clean data.
- Malicious clients participated in every round with corrupted updates.
- The server aggregated updates using both honest and malicious clients.
- The impact of each attack was measured by monitoring test accuracy and loss over multiple rounds.

Our experimental setup provided valuable insights into ClusFed's adaptability under intense adversarial conditions. By dynamically adjusting client selection and clustering mechanisms, ClusFed demonstrated strong resilience against these diverse attack patterns while maintaining high accuracy and stability.

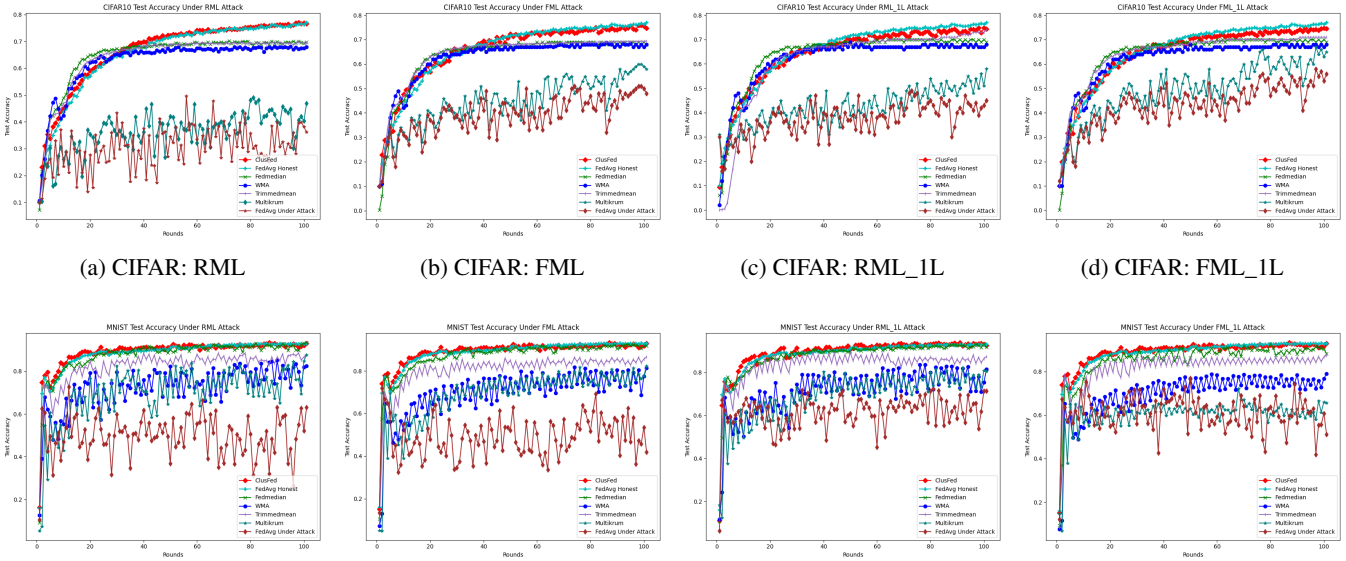
F. EVALUATION METRIC

The primary metric used to evaluate the performance of our ClusFed strategy was global model accuracy. This metric directly reflects the model's ability to make correct predictions on the test set, even amidst Byzantine attacks. Accuracy was measured after each round of FL, allowing us to track performance over time.

Our results revealed that model performance often dipped following attack rounds but showed resilience by recovering in subsequent non-attack rounds. This pattern of attack and recovery is evident in the accuracy curves over the 100 training rounds. The frequent and unpredictable nature of the attacks, combined with a high percentage of malicious clients, created a worst-case scenario that thoroughly tested the robustness of our ClusFed strategy. The strategy's ability to maintain high accuracy, even under these severe conditions, demonstrates its effectiveness in mitigating Byzantine attacks within FL environments.

G. RESULTS AND DISCUSSION

ClusFed was evaluated under various adversarial scenarios, including RML, RML_1L, FML, FML_1L applied to both CIFAR-10 and MNIST datasets (Tables 3 and 4 and Figure 7), and SF for Shakespeare and Water Leak datasets (Figures 5 and 6). These attacks were applied with 40% of clients designated as malicious. The results provide valuable insights into the performance of ClusFed compared to state-of-the-art Byzantine-resilient algorithms such as FedMedian, WMA, TrimmedMean, Multi-Krum, and FedAvg.



(a) CIFAR: RML (b) CIFAR: FML (c) CIFAR: RML_1L (d) CIFAR: FML_1L

(e) MNIST: RML (f) MNIST: FML (g) MNIST: RML_1L (h) MNIST: FML_1L

FIGURE 7: Test Accuracy Over 100 Rounds for Different FL Strategies with CIFAR and MNIST datasets under various attacks (80 Clients, 40% Malicious).

TABLE 3: Comparison of FL Strategies on MNIST for different Attacks

Strategy	**MNIST**							
	RML Acc (%)	**RML Loss**	**RML-1L Acc (%)**	**RML-1L Loss**	**FML Acc (%)**	**FML Loss**	**FML-1L Acc (%)**	**FML-1L Loss**
ClusFed	93.02%	0.843	92.54%	0.890	92.90%	1.234	92.80%	1.223
Fedmedian	93.34%	1.202	91.71%	1.464	91.76%	1.455	88.28%	2.251
WMA	82.57%	3.229	81.38%	3.420	81.28%	3.440	78.99%	3.871
Trimmedmean	87.64%	2.715	87.36%	2.749	86.64%	2.928	87.41%	2.899
Multikrum	87.72%	2.849	80.47%	4.269	82.29%	4.146	65.66%	6.233
FedAvg Under Attack	63.02%	6.543	71.35%	3.879	41.96%	5.062	51.18%	6.532

TABLE 4: Comparison of FL Strategies on CIFAR-10 for different Attacks

Strategy	**CIFAR-10**							
	RML Acc (%)	**RML Loss**	**RML-1L Acc (%)**	**RML-1L Loss**	**FML Acc (%)**	**FML Loss**	**FML-1L Acc (%)**	**FML-1L Loss**
ClusFed	76.63%	2.843	74.32%	3.108	74.68%	3.062	74.55%	3.089
Fedmedian	69.73%	4.217	70.01%	4.154	70.24%	4.126	71.12%	4.053
WMA	67.14%	4.692	68.93%	4.380	68.05%	4.539	69.03%	4.365
Trimmedmean	69.37%	4.081	75.38%	3.215	75.80%	3.146	75.09%	3.230
Multikrum	46.81%	6.341	59.19%	5.247	58.23%	5.340	59.87%	5.160
FedAvg Under Attack	36.16%	7.907	48.06%	6.299	48.55%	6.224	56.70%	5.476

1) Performance on CIFAR-10 Dataset

The CIFAR-10 dataset presents a challenging scenario due to its complex image features and inter-class similarities. Figures 7(a)–(d) illustrate the test accuracy evolution over 100 rounds under the four attack types. ClusFed consistently achieves superior performance compared to other strategies across all attack scenarios.

Under the RML attack (Figure 7(a)), ClusFed achieves a final test accuracy of 76.63%, significantly outperforming FedAvg Under Attack (36.16%) and Multi-Krum (46.81%). This highlights ClusFed's ability to mitigate the widespread noise introduced by random label flipping across all classes. The adaptive client selection mechanism effectively isolates malicious clients, preventing their corrupted updates from dominating the global model.

Under the FML attack (Figure 7(b)), where a fixed offset is applied to all labels, ClusFed maintains robust performance with a final accuracy of 74.68%. In contrast, FedAvg Under Attack achieves only 48.55%, demonstrating its vulnerability to consistent mislabeling patterns. The clustering-based approach in ClusFed ensures that even persistent adversarial behavior is detected and mitigated.

For targeted attacks like RML_1L (Figure 7(c)) and FML_1L (Figure 7(d)), which focus on flipping a single label, ClusFed exhibits remarkable resilience with accuracies of 74.32% and 74.55%, respectively. These results emphasize the effectiveness of ClusFed in handling both global and targeted attacks. By dynamically adapting to evolving adversarial patterns, ClusFed prevents specific class corruption from propagating through the model.

Despite some overlap between honest and malicious clients in CIFAR-10 due to its complex data characteristics, ClusFed's clustering mechanism minimizes their impact by ensuring that a majority of honest clients dominate global

updates. This resilience is crucial for maintaining model integrity in challenging environments.

The radar plot in Figure 8(a) further highlights the trade-offs between test accuracy, training latency, and test loss for CIFAR-10 under RML attacks. ClusFed achieves balanced performance across all metrics, demonstrating its robustness compared to other methods.

2) Performance on MNIST Dataset

The MNIST dataset, with its simpler structure and distinct class features, provides a less challenging environment for FL strategies. Figures 7(e)–(h) show the test accuracy evolution under different attack scenarios. ClusFed consistently outperforms baseline methods, achieving near-optimal accuracy across all attacks.

Under the RML attack (Figure 7(e)), ClusFed achieves a final test accuracy of 93.02%, closely matching FedMedian's performance (93.34%) but significantly outperforming WMA (82.57%) and FedAvg Under Attack (63.02%). The clustering mechanism efficiently isolates malicious clients despite the widespread noise introduced by random label flipping.

For FML attacks (Figure 7(f)), where a fixed offset is applied globally, ClusFed maintains strong performance with an accuracy of 92.90%, demonstrating its resilience against consistent mislabeling patterns.

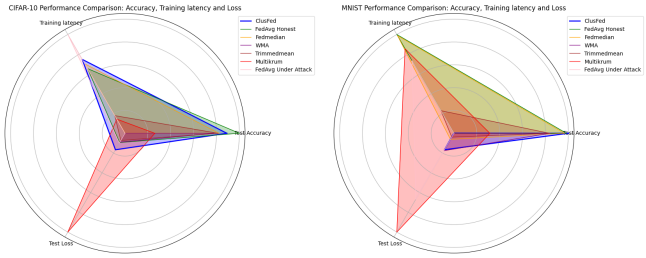
Targeted attacks like RML_1L (Figure 7(g)) and FML_1L (Figure 7(h)) have minimal impact on ClusFed's performance, with final accuracies of 92.54% and 92.80%, respectively. These results indicate that ClusFed's adaptive mechanisms are highly effective in isolating targeted adversarial behavior while preserving overall model integrity.

3) Performance on Shakespeare Dataset

Figure 5 shows the evolution of test accuracy on the Shakespeare next-character task under the SF attack over 100 rounds. Under this extreme gradient inversion, standard FedAvg collapses below 10% accuracy, while robust aggregators such as MultiKrum and WMA plateau around 19% and 29%, respectively. In contrast, ClusFed detects and isolates malicious updates via clustering, recovering steadily to nearly 40% top-1 accuracy by round 100. Trimmed-Mean and FedMedian achieve intermediate performance (approximately 34% and 32%), yet still fall short of ClusFed's resilience. These results demonstrate that ClusFed's variance-penalized clustering effectively separates sign-flipped updates and preserves model utility in sequential language modeling.

4) Performance on Water Leak Dataset

Figure 6 depicts the progression of test accuracy on the Water Leak dataset under the same SF attack. Once again, ClusFed outperforms all other methods, surpassing 85% accuracy by the 100th round. In comparison, honest FedAvg stabilizes around 82%, WMA levels off near 75%, and both Multi-Krum and TrimmedMean reach only about 61%. FedMedian plateaus at approximately 72%, while FedAvg under attack deteriorates below 45%. These results shows how ClusFed has resilience and rapid adaptation in both recovering more quickly and achieving a higher accuracy even on time-series dataset.



(a) CIFAR Radar (b) MNIST Radar

FIGURE 8: Comparing Accuracy, Training Latency, and Loss Across FL Strategies for RML attack.

The radar plot in Figure 8(b) highlights ClusFed's balanced performance across test accuracy, training latency, and test loss for MNIST under RML attacks. Compared to other methods, ClusFed achieves high accuracy with moderate latency, ensuring robust performance even under adversarial conditions.

5) Comparison Across Strategies

Tables 3 and table 4 provide a detailed comparison of test accuracy and loss for various strategies under different attack scenarios on MNIST and CIFAR-10 datasets.

On MNIST: ClusFed achieves comparable or superior accuracy across all attacks while maintaining lower test loss compared to most baseline methods. For example, under FML attacks, ClusFed achieves an accuracy of 92.90% with a loss of 1.234 compared to FedMedian's accuracy of 91.76% with a higher loss of 1.455. The high accuracy achieved by FedMedian highlights its robustness in simpler datasets like MNIST; however, its higher loss values indicate less efficient convergence compared to ClusFed.

On CIFAR-10: ClusFed consistently outperforms other strategies in terms of both accuracy and loss. Under RML attacks, ClusFed achieves an accuracy of 76.63% with a loss of 2.843 compared to FedMedian's accuracy of 69.73% with a higher loss of 4.217. Multi-Krum performs poorly across all attack scenarios due to its vulnerability to non-IID data distributions inherent in FL setups.

These results highlight the importance of adaptive client selection mechanisms in FL systems when facing diverse attack strategies.

CIFAR-10's complex features make it more susceptible to adversarial perturbations compared to MNIST, with targeted attacks like FML_1L having a greater impact due to inter-class similarities that amplify label flipping effects. ClusFed's clustering-based client selection mechanism dynamically adapts to evolving adversarial patterns, ensuring robust performance across both global (RML/FML) and targeted

(RML_1L/FML_1L) attacks. While baseline methods like FedMedian perform well on simpler datasets like MNIST, they struggle on more complex datasets like CIFAR-10. Multi-Krum consistently underperforms across both datasets due to its sensitivity to non-IID data distributions, further emphasizing the importance of adaptive mechanisms like those employed by ClusFed.

Clustering Effectiveness

The effectiveness of ClusFed in distinguishing between honest and malicious clients is evident through its adaptive clustering mechanism. This approach dynamically adjusts to varying participation levels and attack scenarios, ensuring robust performance. In environments with full client participation, ClusFed effectively groups most honest clients together, even though some overlap with malicious clients may occur due to complex data characteristics. However, this overlap is minimized by the dominance of honest client updates, which maintain model integrity. In scenarios with reduced client participation, ClusFed's clustering becomes more precise, clearly separating honest and malicious clients. This adaptability highlights ClusFed's capability to enhance robustness in FL environments by dynamically responding to changes in client behavior and participation. The results demonstrate that ClusFed's clustering mechanism consistently identifies and isolates adversarial influences across both CIFAR-10 and MNIST datasets. Despite the inherent complexity of CIFAR-10, which presents challenges due to inter-class similarities, ClusFed maintains high accuracy by ensuring that honest clients dominate global updates. On the simpler MNIST dataset, ClusFed achieves perfect separation of client types, reflecting its effectiveness in environments with distinct class features. These findings underscore the potential of ClusFed to achieve high accuracy and resilience in FL systems. By leveraging adaptive clustering, ClusFed provides a robust defense against diverse adversarial strategies without relying on rigid assumptions or static thresholds.

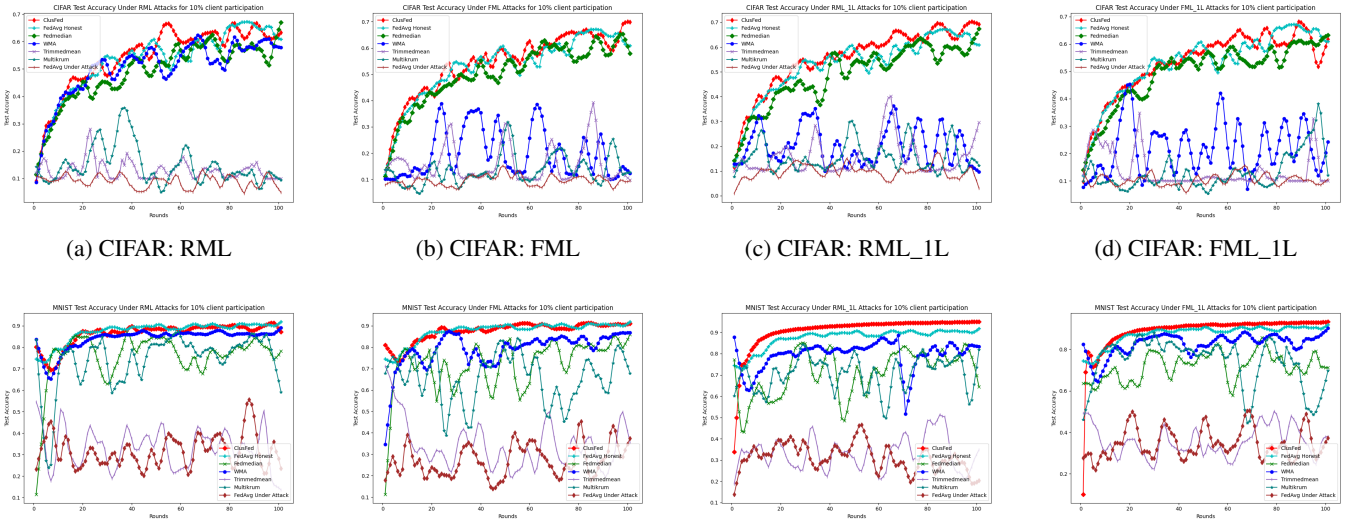
6) Partial Client Participation

To demonstrate ClusFed's practical viability under realistic deployment constraints, we conducted all experiments shown in Figures 9(a)–(d) and (e)–(h) with only 10% of clients participating in each training round. A more realistic federated scenario where full participation is rarely feasible. FL typically operates under partial availability due to limited device uptime and network variability. By clustering client updates and selecting a small, coherent subset, ClusFed reduces per-round communication and computation overhead by approximately 90% while preserving convergence speed and accuracy. On CIFAR-10, ClusFed achieved 70% accuracy by round 30 and stabilized above 75% by round 100, similar performance to full-participation FedAvg by round 60, representing a 50% reduction in training time ($p<0.01$). On MNIST, ClusFed reached 90% accuracy by round 25 and plateaued at 92.7% by round 80. Even under this stringent setting, ClusFed rapidly recovers and maintains high accuracy across different attack scenarios for both MNIST and CIFAR10, while requiring fewer resources per round. These results highlight how adaptive client selection focuses computation on the most informative and honest participants, mitigating gradient noise from stragglers and adversaries, and ensuring robust model updates in non-IID settings.

VI. OPEN CHALLENGES AND FUTURE DIRECTIONS

While the proposed ClusFed strategy demonstrates robust performance against Byzantine attacks, several open challenges remain in FL that warrant further investigation. Addressing these challenges will not only enhance the security and scalability of FL systems but also broaden their applicability across diverse domains. The current work focuses primarily on label-flipping attacks and sign-flipping attacks, a well-studied category of adversarial threats in FL. However, other sophisticated attack vectors, such as evasion attacks [33], membership inference attacks [65], and generative adversarial network (GAN)-based attacks pose significant risks to FL systems. For instance, evasion attacks target the model at inference time by crafting adversarial inputs that induce misclassification without altering model weights. These attacks are particularly challenging to detect at inference time, since the adversarial inputs are crafted to appear indistinguishable from benign samples and thus do not significantly affect overall accuracy while still causing targeted misclassifications. Similarly, GAN-based attacks can generate synthetic data that mimics real distributions to manipulate FL models or extract sensitive information [66], [67]. Investigating the resilience of ClusFed against such advanced attack scenarios is an important direction for future research. Additionally, scaling ClusFed to larger networks introduces challenges related to communication overhead, client heterogeneity, and dynamic participation. Real-world FL deployments often involve thousands or even millions of clients, such as in IoT ecosystems or large-scale healthcare networks [9], [68]. Future work should focus on optimizing ClusFed's clustering mechanism to handle large-scale networks efficiently while maintaining its robustness against adversarial threats. While ClusFed has shown robust defense against Byzantine threats across image, text, and time-series benchmarks, its evaluation is still limited to common classification tasks on MNIST, CIFAR-10, Shakespeare. Real-world FL applications often involve more complex domains, such as continuous speech recognition and language modeling [69] and language modeling [70], which require richer architectures (e.g., CNN-LSTM hybrids, transformer-based models) and specialized evaluation metrics (e.g., word error rate for audio, BLEU for text). Moreover, the choice of metric can greatly influence perceived robustness, so future work will investigate task-specific criteria and calibration methods. We will also explore scaling ClusFed to deeper networks and larger client populations for a realistic federated environments. [71], [72].

Testing ClusFed on such datasets will help assess its generalizability across different domains. For example, applying ClusFed in NLP tasks like sentiment analysis or machine



(a) CIFAR: RML (b) CIFAR: FML (c) CIFAR: RML_1L (d) CIFAR: FML_1L

(e) MNIST: RML (f) MNIST: FML (g) MNIST: RML_1L (h) MNIST: FML_1L

FIGURE 9: Test Accuracy Over 100 Rounds for Different FL Strategies with CIFAR and MNIST datasets under various attacks For **10% of Partial Participation** of Clients(80 Clients, 40% Malicious).

translation could reveal new insights into its performance under adversarial conditions. Most existing studies in FL, including this work, primarily evaluate model performance using accuracy. However, other metrics such as precision, recall, F1-score, and robustness against fairness violations are equally important in assessing the efficacy of FL systems [21], [26]. Incorporating these metrics into future evaluations will provide a more holistic understanding of ClusFed's performance. Additionally, measuring computational efficiency and energy consumption could offer insights into its practicality for resource-constrained environments like IoT networks. While ClusFed focuses on robustness against adversarial attacks, privacy concerns remain a critical aspect of FL. Techniques such as DP and secure multi-party computation have been widely explored to protect client data during training [66], [73]. Future research could integrate these techniques into ClusFed to enhance its privacy guarantees without compromising its robustness against adversarial threats. Finally, transitioning from experimental setups to real-world deployments requires addressing standardization challenges in FL. Issues such as interoperability between different FL frameworks, compliance with data protection regulations (e.g., GDPR), and integration with existing infrastructure need to be considered [71], [72]. Collaborations with industry stakeholders could accelerate the adoption of robust FL strategies like ClusFed in practical applications.

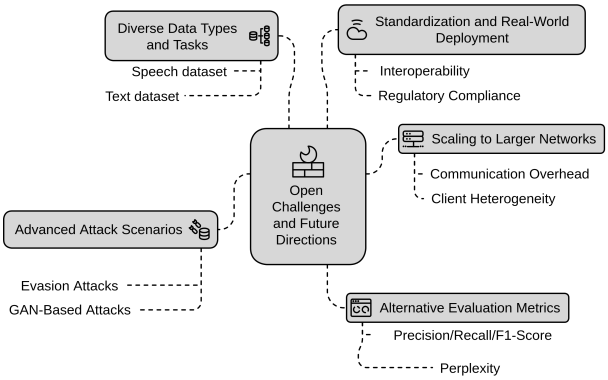


FIGURE 10: Open research challenges and potential future directions for Federated Learning security.

VII. CONCLUSION

This study has introduced ClusFed, a pioneering strategy designed to bolster the resilience of FL systems against Byzantine attacks. By integrating adaptive client selection mechanisms with clustering techniques, ClusFed effectively differentiates between honest and malicious clients, thereby safeguarding the integrity and performance of the global model. This approach represents a significant advancement in FL by addressing the limitations of static defenses and rigid assumptions prevalent in existing methods. ClusFed's ability to dynamically adapt to evolving adversarial patterns ensures robust performance across diverse environments, making it particularly suited for non-IID data settings common in real-world applications. The strategy's innovative use of clustering allows it to isolate malicious influences without compromising the contributions of honest clients, thereby maintaining high model accuracy and stability. This adaptability is crucial in environments characterized by heterogeneous data distributions and varying client participation rates. The implications of this work extend beyond immediate security enhancements; they highlight the potential for developing more sophisticated FL systems capable of withstanding com-

plex adversarial threats. ClusFed's design not only enhances security but also lays the groundwork for future innovations in FL, emphasizing the importance of dynamic adaptation and robust client selection. Looking forward, several avenues for future research are evident. Further exploration into advanced attack vectors, such as adversarial robustness unhardening, could provide deeper insights into ClusFed's efficacy. Additionally, scaling ClusFed to larger networks and testing its applicability across various data types and tasks will be critical in assessing its generalizability. Incorporating alternative evaluation metrics beyond accuracy will offer a more comprehensive understanding of its performance. In conclusion, ClusFed represents a promising pathway toward developing resilient distributed learning systems that can effectively navigate the complexities of modern adversarial landscapes. By continuing to address open challenges in this domain, ClusFed can pave the way for more secure and adaptable FL deployments capable of meeting the demands of increasingly sophisticated threats.

• • •

REFERENCES

[1] S. Baïna and A. Rihi, "New Insight of Data Mining-Based Fraud Detection in Financial IT-Systems," in 15th IADIS International Conference Information Systems 2022, 2022.

[2] A. Harbouche, M. Erradi, and A. Kobbane, "A flexible wireless body sensor network system for health monitoring," in 2013 Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE). IEEE, 2013, pp. 44–49.

[3] A. Walid, M. El Kamili, A. Kobbane, A. Mabrouk, E. Sabir, and M. El Koutbi, "A decentralized network selection algorithm for group vertical handover in heterogeneous networks," in 2014 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, 2014, pp. 2817–2821.

[4] A. Rihi, S. Baïna, F.-Z. Mhada, E. El Bachari, H. Tagemouati, M. Guerboub, I. Benzakour, K. Baïna, and E. H. Abdelwahed, "Innovative predictive maintenance for mining grinding mills: From LSTM-based vibration forecasting to pixel-based MFCC image and CNN," The International Journal of Advanced Manufacturing Technology, vol. 135, no. 3, pp. 1271–1289, November 2024.

[5] S. Lhazmir, O. A. Oualhaj, A. Kobbane, and J. Ben-Othman, "Matching game with no-regret learning for iot energy-efficient associations with uav," IEEE Transactions on Green Communications and Networking, vol. 4, no. 4, pp. 973–981, 2020.

[6] M.-A. Koulali, S. Koulali, H. Tembine, and A. Kobbane, "Industrial internet of things-based prognostic health management: A mean-field stochastic game approach," IEEE Access, vol. 6, pp. 54 388–54 395, 2018.

[7] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Aguera y Arcas, "Communication-efficient learning of deep networks from decentralized data," in Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), vol. 54. PMLR, 2017, pp. 1273–1282.

[8] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, " A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection ," IEEE Transactions on Knowledge & Data Engineering, vol. 35, no. 04, pp. 3347–3366, 2023.

[9] N. Rieke, J. Hancox, W. Li, F. Milletari, H. R. Roth, S. Albarqouni, S. Bakas, M. N. Galtier, B. A. Landman, K. H. Maier-Hein, S. Ourselin, M. J. Sheller, R. M. Summers, A. Trask, D. Xu, M. Baust, and M. J. Cardoso, "The future of digital health with federated learning," NPJ Digital Medicine, vol. 3, no. 1, pp. 1–7, 2020.

[10] M. J. Sheller, B. Edwards, G. A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, Z. Xu, D. S. Marcus, R. R. Colen et al., "Federated learning in medicine: Facilitating multi-institutional collaborations without sharing patient data," Scientific Reports, vol. 10, no. 1, pp. 1–12, 2020.

[11] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," 2018, arXiv preprint arXiv:1811.03604. [Online]. Available: https://arxiv.org/abs/1811.03604

[12] L. Lyu, H. Yu, J. Zhao, and Q. Yang, "Threats to federated learning," in Federated Learning: Privacy and Incentive, Q. Yang, L. Fan, and H. Yu, Eds. Cham, Switzerland: Springer International Publishing, 2020, pp. 3–16.

[13] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, ser. Proceedings of Machine Learning Research, S. Chiappa and R. Calandra, Eds., vol. 108. PMLR, 26–28 Aug 2020, pp. 2938–2948.

[14] F. A. Yerlikaya and Şerif Bahtiyar, "Data poisoning attacks against machine learning algorithms," Expert Systems with Applications, vol. 208, p. 118101, 2022.

[15] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, "Data poisoning attacks against federated learning systems," in Computer Security – ESORICS 2020: 25th European Symposium on Research in Computer Security, ser. Lecture Notes in Computer Science, L. Chen, N. Li, K. Liang, and S. Schneider, Eds., vol. 12309. Cham: Springer International Publishing, 2020, pp. 480–501.

[16] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in 2016 IEEE European Symposium on Security and Privacy (EuroS&P), 2016, pp. 372–387.

[17] S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane, "Adversarial attacks on medical machine learning," Science, vol. 363, no. 6433, pp. 1287–1289, 2019.

[18] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning models," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018, pp. 1625–1634.

[19] M. Fang, X. Cao, J. Jia, and N. Gong, "Local model poisoning attacks to Byzantine-Robust federated learning," in 29th USENIX Security Symposium (USENIX Security 20). USENIX Association, Aug. 2020, pp. 1605–1622.

[20] Z. Tian, L. Cui, J. Liang, and S. Yu, "A comprehensive survey on poisoning attacks and countermeasures in machine learning," ACM Computing Surveys (ACM Comput. Surv.), vol. 55, no. 8, Dec. 2022.

[21] L. Ni, X. Gong, J. Li, Y. Tang, Z. Luan, and J. Zhang, "rfedfw: Secure and trustable aggregation scheme for byzantine-robust federated learning in internet of things," Information Sciences, vol. 653, p. 119784, 2024.

[22] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," IEEE Signal Processing Magazine, vol. 37, no. 3, pp. 50–60, 2020.

[23] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: byzantine tolerant gradient descent," in Proceedings of the 31st International Conference on Neural Information Processing Systems, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 118–128.

[24] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in Proceedings of the 35th International Conference on Machine Learning, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 5650–5659.

[25] E. M. El Mhamdi, S. Farhadkhani, R. Guerraoui, A. H. A. Guirguis, L. N. Hoang, and S. L. A. Rouault, "Collaborative learning in the jungle (decentralized, byzantine, heterogeneous, asynchronous and nonconvex learning)," in Advances in Neural Information Processing Systems 34 pre-proceedings (NeurIPS 2021), ser. Advances in Neural Information Processing Systems; 34, dec 2021.

[26] S. Li, Y. Cheng, W. Wang, Y. Liu, and T. Chen, "Learning to detect malicious clients for robust federated learning," 2020, arXiv preprint. [Online]. Available: https://arxiv.org/abs/2002.00211

[27] Q. Shen, P. Shi, J. Zhu, S. Wang, and Y. Shi, "Neural networks-based distributed adaptive control of nonlinear multiagent systems," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 3, pp. 1010–1021, 2020.

[28] T. Wang, Z. Zheng, and F. Lin, "Federated learning framework based on trimmed mean aggregation rules," Expert Systems with Applications, vol. 270, p. 126354, 2025.

[29] K. Otmani, R. El-Azouzi, and V. Labatut, "Fedsv: Byzantine-robust federated learning via shapley value," in ICC 2024 - IEEE International Conference on Communications, 2024, pp. 4620–4625.

[30] J. Kang, Z. Xiong, D. Niyato, H. Yu, Y.-C. Liang, and D. I. Kim, "Incentive design for efficient federated learning in mobile networks: A contract theory approach," in 2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS), 2019, pp. 1–5.

[31] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-i.i.d. data," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 9, pp. 3400–3413, 2020.

[32] S. Wang, J. Hayase, G. Fanti, and S. Oh, "Towards a defense against federated backdoor attacks under continuous training," arXiv preprint arXiv:2205.11736, 2023, last revised 31 Jan 2023. [Online]. Available: https://arxiv.org/abs/2205.11736

[33] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, "Analyzing federated learning through an adversarial lens," in Proceedings of the 36th International Conference on Machine Learning, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 634–643.

[34] C. Yang, Q. Wu, and Y. Chen, "Generative poisoning attack method against neural networks," 2017, arXiv preprint. [Online]. Available: https://arxiv.org/abs/1703.01340

[35] Y. Li, Y. Jiang, Z. Li, and S.-T. Xia, "Backdoor learning: A survey," IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 1, pp. 5–22, 2024.

[36] Z. C. Ala Gouissem and R. Hamila, "A comprehensive survey on client selections in federated learning," in Innovation and Technological Advances for Sustainability: Proceedings of the International Conference on Innovation and Technological Advances for Sustainability (ITAS 2023). Boca Raton, FL, USA: CRC Press, Taylor & Francis Group, 2024, pp. 417–428.

[37] F. Lai, X. Zhu, H. V. Madhyastha, and M. Chowdhury, "Efficient federated learning via guided participant selection," in USENIX Symposium on Operating Systems Design and Implementation (OSDI), 2021.

[38] B. Zhu, L. Wang, Q. Pang, S. Wang, J. Jiao, D. Song, and M. I. Jordan, "Tackling the objective inconsistency problem in heterogeneous federated optimization," in Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS), ser. Proceedings of Machine Learning Research, vol. 206. PMLR, 2023, pp. 3151–3178.

[39] M. Xhemrishi, J. Östman, A. Wachter-Zeh, and A. Graell i Amat, "FedGT: Identification of malicious clients in federated learning with secure aggregation," 2023.

[40] Z. Zhang, X. Cao, J. Jia, and N. Z. Gong, "Fldetector: Defending federated learning against model poisoning attacks via detecting malicious clients," in Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, ser. KDD '22. New York, NY, USA: ACM, 2022, p. 2545–2555.

[41] X. Cao, Z. Zhang, J. Jia, and N. Z. Gong, "Flcert: Provably secure federated learning against poisoning attacks," IEEE Transactions on Information Forensics and Security, vol. 17, pp. 3691–3705, 2022.

[42] E. Isik-Polat, G. Polat, and A. Koçyiğit, "Arfed: Attack-resistant federated averaging based on outlier elimination," Future Gener. Comput. Syst., vol. 141, pp. 626–650, 2021.

[43] K. Pillutla, S. M. Kakade, and Z. Harchaoui, "Robust aggregation for federated learning," IEEE Transactions on Signal Processing, vol. 70, pp. 1142–1154, 2022.

[44] X. Gong, Y. Chen, Q. Wang, and W. Kong, "Backdoor attacks and defenses in federated learning: State-of-the-art, taxonomy, and future directions," IEEE Wireless Communications, vol. 30, no. 2, pp. 114–121, 2023.

[45] T. D. Nguyen, P. Rieger, H. Chen, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, S. Zeitouni, F. Koushanfar, A.-R. Sadeghi, and T. Schneider, "FLAME: Taming backdoors in federated learning," in 31st USENIX Security Symposium (USENIX Security 22). Boston, MA: USENIX Association, Aug. 2022, pp. 1415–1432.

[46] P. Qi, D. Chiaro, A. Guzzo, M. Ianni, G. Fortino, and F. Piccialli, "Model aggregation techniques in federated learning: A comprehensive survey," Future Generation Computer Systems, vol. 150, pp. 272–293, 2024.

[47] J. Kim, C. Song, J. Paek, J.-H. Kwon, and S. Cho, "A review on research trends of optimization for client selection in federated learning," in 2024 International Conference on Information Networking (ICOIN), 2024, pp. 287–289.

[48] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, ser. CCS '17. New York, NY, USA: ACM, 2017, p. 1175–1191.

[49] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client-level perspective," arXiv preprint arXiv:1712.07557, 2017. [Online]. Available: https://arxiv.org/abs/1712.07557

[50] S. Truex, C. Mobley, P. Khozeimeh, and et al., "Ldp-fed: Federated learning with local differential privacy," Proceedings on Privacy Enhancing Technologies, vol. 2019, no. 4, pp. 167–184, 2019.

[51] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in ICC 2019 - 2019 IEEE International Conference on Communications (ICC). IEEE, 2019, pp. 1–7.

[52] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Optimizing federated learning on non-iid data with reinforcement learning," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 5, pp. 1867–1879, 2020.

[53] H. Kabbaj, R. El-Azouzi, and A. Kobbane, "Robust federated learning via weighted median aggregation," in 2024 2nd International Conference on Federated Learning Technologies and Applications (FLTA). IEEE, 2024, pp. 298–303.

[54] H. Kabbaj, M. El Hanjri, A. Kobbane, R. El Azouzi, and A. Abouaomar, "Distfl: An enhanced fl approach for non trusted setting in water distribution networks," in Proceedings of the 2023 IEEE International Conference on Communications (ICC). IEEE, 2023, pp. 2486–2491.

[55] S. P. Lloyd, "Least squares quantization in pcm," IEEE Transactions on Information Theory, vol. 28, no. 2, pp. 129–137, 1982.

[56] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, J. Fernandez-Marques, Y. Gao, L. Sani, H. L. Kwing, T. Parcollet, P. P. d. Gusmão, and N. D. Lane, "Flower: A friendly federated learning research framework," arXiv preprint arXiv:2007.14390, 2020.

[57] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," in Workshop on Federated Learning for Data Privacy and Confidentiality in Conjunction with NeurIPS 2019, 2019, pp. 1–4.

[58] J. V. D. Walt, P. Heyns, and D. Wilke, "Single section of pipe for leak detection," 9 2020. [Online]. Available: https://figshare.com/articles/dataset/Single_section_of_pipe_for_leak_detection/12973148

[59] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," in 2022 IEEE 38th International Conference on Data Engineering (ICDE), 2022, pp. 965–978.

[60] N. M. Jebreel, J. Domingo-Ferrer, D. Sánchez, and A. Blanco-Justicia, "Lfighter: Defending against the label-flipping attack in federated learning," Neural Networks, vol. 170, pp. 111–126, 2024.

[61] N. M. Jebreel, J. Domingo-Ferrer, D. Sánchez, and A. Blanco-Justicia, "Defending against the label-flipping attack in federated learning," Information Sciences, vol. 610, pp. 123–140, 2022.

[62] T. Kim, J. Li, N. Madaan, S. Singh, and C. Joe-Wong, "Adversarial robustness unhardening via backdoor attacks in federated learning," arXiv preprint arXiv:2310.11594, 2023.

[63] L. Lavaur, Y. Busnel, and F. Autrel, "Systematic analysis of label-flipping attacks against federated learning in collaborative intrusion detection systems," in Proceedings of the 19th International Conference on Availability, Reliability and Security, ser. ARES '24. New York, NY, USA: Association for Computing Machinery, 2024.

[64] S. Li, Y. Cheng, Y. Liu, W. Wang, and T. Chen, "Abnormal client behavior detection in federated learning," https://arxiv.org/abs/1910.09933, 2019, preprint (arXiv:1910.09933).

[65] 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). Los Alamitos, CA, USA: IEEE Computer Society, May 2017, pp. 3–18.

[66] J. Hasan, "Security and privacy issues of federated learning," arXiv preprint arXiv:2307.12181, 2023.

[67] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the gan: Information leakage from collaborative deep learning," in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, ser. CCS '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 603–618.

[68] V. Stamatis, P. I. Radoglou-Grammatikis, A. Sarigiannidis, N. Pitropakis, T. Lagkas, V. Argyriou, E. K. Markakis, and P. G. Sarigiannidis, "Advancements in federated learning for health applications: A concise survey," in Proceedings of DCOSS-IoT 2024 (International Conference on

Distributed Computing in Sensor Systems and Internet of Things), 2024, pp. 503–508.
[69] D. Amodei, R. Anubhai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh et al., "Deep Speech 2: End-to-end speech recognition in english and mandarin," in Proceedings of the 33rd International Conference on Machine Learning (ICML), 2016, pp. 173–182.
[70] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in Neural Information Processing Systems, vol. 30, 2017, pp. 5998–6008.
[71] C. Papadopoulos, K.-F. Kollias, and G. F. Fragulis, "Recent advancements in federated learning: State of the art, fundamentals, principles, iot applications and future trends," Future Internet, vol. 16, no. 11, 2024.
[72] D. C. Nguyen, M. Ding, P. N. Pathirana, A. P. Seneviratne, J. Li, and F. I. H. V. Poor, "Federated learning for internet of things: A comprehensive survey," IEEE Communications Surveys & Tutorials, vol. 23, pp. 1622–1658, 2021.
[73] X. Yin, Y. Zhu, and J. Hu, "A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions," ACM Comput. Surv., vol. 54, no. 6, Jul. 2021.

HIBATALLAH KABBAJ received her M.S. degree in data science and big data from the National School of Computer Science and Systems Analysis (ENSIAS), Rabat, Morocco. Currently pursuing a Ph.D. degree in computer science at Avignon Université, France, and ENSIAS in Morocco, under the supervision of Prof. Rachid Elazouzi and Prof. Abdellatif Kobbane. Her Ph.D. research focuses on security in federated learning, specifically on detecting malicious clients within decentralized learning systems. Her research interests include machine learning, federated learning, adversarial machine learning, anomaly detection, and privacy-preserving artificial intelligence.

RACHID ELAZOUZI is a full professor at the University of Avignon. He received his PhD in Applied Mathematics from Mohammed V University in 2000. He joined the National Institute for Research in Computer Science and Control (INRIA), where he held positions as a postdoctoral fellow and research engineer. In 2003, he joined the University of Avignon as an associate professor. He was a visitor scholar in the Department of Computer Science at the University of California at Berkeley in 2012-2013 and in the Department of Computer Science at Carnegie Mellon University in 2023-2024. From 2016 to 2023, he was director of the research federation (CNRS- FR 3621 Agorantic) comprising 9 laboratories covering 100 faculties, focused on interdisciplinary research on social networks. He was an AE of the IEEE TON journal 2016-2020 and he is an AE of the IEEE TNSE (2021-2023) and Editor of the IEEE TNSE (2024-now). His research interests include networking games, economic networks, resource allocation, social networks, wireless networks, complex systems, machine learning and performance evaluation.

ABDELLATIF KOBBANE is a Full Professor at the Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS), Mohammed V University in Rabat, Morocco, since 2009. He received his M.S. in Computer Science, Telecommunication, and Multimedia from Mohammed V-Agdal University in 2003 and his Ph.D. from Mohammed V-Agdal University and the University of Avignon, France. His research focuses on wireless mobile networking, performance evaluation, flexible resource management, and distributed AI in 5G/6G networks, leveraging AI and mean-field game theory. He has authored numerous publications in leading IEEE conferences and journals, including IEEE ICC, IEEE Globecom, IWCMC, ICNC, and IEEE WCNC. His work extends to Digital Twins, IoT, SDN/NFV, blockchain, trust and cybersecurity, mobile cloud computing, and emerging technologies. He also explores AI and Generative AI applications in agriculture, healthcare, natural resource management, and societal well-being. Dr. Kobbane is a Senior Member of IEEE ComSoc, former Chair of the IEEE Communication Software Technical Committee, and the Founder of the WINCOM conference. He also established the Association of Research in Mobile Wireless Networks and Embedded Systems (MobiTic) in Morocco and previously led the Master's program in IoT and Mobile Services (IOSM).