STARFed: Link-Aware Defense Against Poisoning Attacks in Satellite-Terrestrial Federated Learning

Zizheng Liu, Bharat K. Bhargava, *Life Fellow, IEEE*, and Nagender Aneja

***Abstract*—Satellite-ground integrated computation where machine learning models trained on satellites and aggregated on Earth offers novel opportunities for federated learning (FL). While satellites in space provide isolated computing environments, satellite-terrestrial (S-T) communication links are exposed to spoofing and hijacking attacks, making transmitted models vulnerable to poisoning attacks. To address this paradigm-specific threat, we introduce STARFed, a novel framework that enhances robustness of satellite-based FL by leveraging S-T link characteristics during model transmission. It comprises three components: (1) crowdsourcing-based link authentication, (2) hybrid poison model detection based on both S-T link and model characteristics, and (3) reputation-based model filtering against adaptive adversaries. Our link-aware defense is of independent interest and can be combined with various FL robust aggregation schemes. We evaluate the framework's resilience through comprehensive experiments spanning five dataset-model settings and five attacks, including both model and data poisoning attacks. The framework's performance is compared with six state-of-the-art robust FL aggregation schemes in scenarios with varying degrees of non-IID data distribution, client dropout, and adversarial participation. STARFed demonstrates robust performance across all test scenarios, standing as the only defense mechanism to maintain effectiveness throughout. In the most favorable case, it achieves an increase in FL accuracy of 15.6% compared to the best link-unaware aggregation scheme, with minimal overhead introduced.**

***Index Terms*—Federated learning, satellite-terrestrial integrated computing, poisoning attacks, crowdsourcing, reputation systems.**

I. Introduction

Space-based computation is opening new frontiers in artificial intelligence. SpaceX's recent Transporter 11 mission [1, 2], which deployed NVIDIA AI GPUs into space, demonstrates the growing industry commitment to satellite-based AI computing. The space environment offers unique advantages for AI training: efficient cooling and continuous solar power provide sustainable energy solutions, while satellites benefit from enhanced security through their physical isolation and controlled access points [3, 4]. Within this evolving space-computation landscape, Federated Learning (FL) emerges as an ideal paradigm where a ground-based coordinator orchestrates the global model while leveraging satellites' training resources and environments for local training. This distributed approach not only unleashes space-exclusive resources but also avoids difficulties in space-ground data sharing, including privacy requirements [5], regulatory compliance [6], and communication bandwidth limitations [7].

Traditional FL systems face long-concerned vulnerabilities to poisoning attacks, where compromised clients inject poisoned data (during training) or models (during aggregation) to undermine training. However, space-based FL exposes new security challenges that demand rethinking of the conventional threat models. Unlike many existing FL systems involving clients such as commercial off-the-shelf (COTS) mobile phones or IoT devices that can be easily controlled or compromised by malicious users [8], satellites provide naturally isolated, difficult-to-compromise training environments or trusted execution environments (TEEs) [3, 9]. First, satellites are more difficult to physically access after being launched into space, greatly increasing the attack threshold and cost. Second, satellites used to train machine learning models can provide a TEE through the deployment of specialized security hardware [3], which is uncommon in COTS mobile phones and IoT devices. Successful examples include the deployment of security-critical applications on satellites such as hardware security modules for root-of-trust storage [3], blockchain consensus mechanisms [10], quantum key distribution [11], and randomness generation [12], which are all enabled by satellites' physically isolated and tamper-resistant nature.

Despite enhanced onboard training, satellite-terrestrial (S-T) links remain the system's Achilles' heel, vulnerable to spoofing and hijacking attacks [13, 14, 15] due to long distances, varying signal strengths, and environmental interference. This inverted security paradigm, where endpoints are more secure than communication channels, represents a blind spot in current satellite-based FL research, necessitating a shift in how we ensure FL security in space-ground systems. Motivated, we propose a novel resilient FL framework specifically designed for the unique characteristics of such systems. Current research exhibits a critical gap that existing satellite-based FL studies [16, 17, 18, 19, 20, 21] overlook their security aspects, while conventional FL security mechanisms [22, 23, 24, 25, 26] fail to account for the unique security challenges in space-ground systems.

To bridge this gap, we introduce STARFed: Satellite-Terrestrial Authenticated Resilient Federated Learning, a novel framework that leverages S-T link characteristics to authenticate model transmissions. Our architecture as illustrated in Figure 1 employs ground relays in a dual role: forwarding models between satellites and the server while collecting S-T link status information to validate transmission authen-

Zizheng Liu and Bharat K. Bhargava are with the Department of Computer Science, Purdue University, West Lafayette, IN 47906, USA. E-mail: {lzz, bbshail}@purdue.edu.

Nagender Aneja is with Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061, USA. E-mail: naneja@vt.edu

ticity. This topology is supported from both standard and practical perspectives. According to 3GPP 5G NR Non-Terrestrial Networks (NTN) [27, 28], ground relays should connect the satellite payload to the terrestrial core (e.g., FL aggregation server) via feeder links, and the satellite should be able to switch among relays to ensure continuity. Meanwhile, the market of satellite-capable devices such as mobile phones and IoT modules is growing: mainstream smartphones and modules now support satellite connectivity (e.g., Apple iPhone 14+ [29], AT&T/AST SpaceMobile 5G calls [30], Starlink [31]/T-Mobile [32] direct-to-cell texting), while 3GPP Rel-17 has standardized NB-IoT/eMTC over NTN for massive IoT [33]. Such a topology guarantees participation under intermittent links [34, 35] and achieves ubiquitous connectivity through dense relay deployments [36, 37, 38, 39], as already demonstrated in operational space networks. However, S-T link status reported by relays does not come free, it introduces a new security consideration: potentially malicious ground relays may report falsified link information. This challenge is further compounded by space-ground communication issues, including client dropout due to unstable S-T links [7] and non-IID data distribution [40], undermining the overall training robustness.

STARFed addresses these challenges through three integrated components: 1) A crowdsourcing-based link authentication system that validates S-T link legitimacy by collecting and cross-validating physical link characteristics reported by multiple ground relays; 2) A hybrid poison model detection approach that combines validated link information with model behavior analysis to identify poisoned models; 3) A reputation-based model filter that tracks relay trustworthiness over time, enforcing adaptive adversaries to contribute more benign models than poisoned ones to maintain positive reputations. Through the coordinated operation of its components, STARFed achieves robust model aggregation under untrusted S-T links and ground relays. Moreover, we believe integrating link status awareness is independently useful for otherwise link-unaware FL robust aggregation frameworks. It provides an additional source for determining model qualifications or weights under various aggregation schemes. We also noticed that, regardless of isolation, the satellites can still be compromised. Our experiments show that even when both satellites and S–T links are compromised by colluding adversaries, STARFed tolerates a corruption ratio of at least 60%.

To the best of our knowledge, our work proposes the first comprehensive approach to secure FL that consider the unique threat model of space-ground systems while maintaining robustness against client dropouts and non-IID data challenges. Our main contributions include:

- A novel framework, STARFed, that leverages unreliable S-T link information for a robust satellite-based FL aggregation scheme, exploring how model transmission channel characteristics can enhance FL training robustness.
- Security analysis for the robustness of the hybrid poison model detector and reputation-based filter, as well as the integrated analysis for the framework's overall sensitivity to adaptive adversaries and unpredictable S-T link conditions.
- Comprehensive experimental evaluation across five dataset-model combinations, five attack types, varying degrees of non-IID data distribution, and diverse dropout and adversarial scenarios, demonstrating STARFed's superior robustness compared to state-of-the-art robust FL aggregation schemes across all settings.

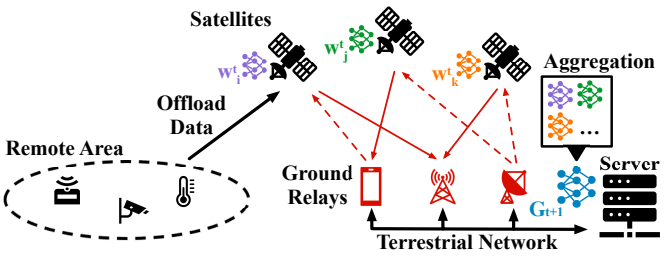


Fig. 1: System architecture of STARFed.

The rest of the paper is organized as follows: We begin with a review of related works in Section II. Section III introduces the preliminaries of FL and its poisoning attacks. Section IV outlines the system and threat models, followed by the problem definition. In Section V, we provide an in-depth exploration of STARFed's key components. Section VI presents the theoretical analysis of STARFed's robustness and the evaluation results are demonstrated in Section VII. Finally, we conclude the paper with Section VIII.

II. Related Work

FL security has been studied since its introduction by McMahan et al. [41], with early works such as [26] and [25] providing robustness under minimal assumptions. Later defenses addressed practical threats with more relaxed assumptions, including FLTrust [24] for trust-bootstrapped aggregation, BaFFLe [23] for client-side cross-validation, FLAME [22] for hybrid defense solutions, and FL-Guardian [42] for layer-wise backdoor mitigation. In parallel, FL has recently been applied in satellite contexts, with works such as [7], [17], and [19] optimizing bandwidth efficiency, scalability, and satellite–ground coordination to achieve faster model convergence.

Meanwhile, the security of satellite communications has long been a concern, particularly regarding spoofing attacks. Studies such as [13], [14], and [15] have revealed systemic vulnerabilities in satellite mega-constellations as well as concrete overshadowing attacks against downlinks. FL emerges as a natural solution to safeguard such tasks, as demonstrated in [43] and [44], which apply FL to various satellite-involved tasks to enhance security and safety.

However, the security of FL itself in satellite contexts has received little attention. Our work addresses this gap by designing defenses that ensure practical robustness under vulnerable satellite–terrestrial links. The most relevant work is SFL-LEO [45], which protects the FL training process on satellite networks. However, it is limited to a topology where low-Earth orbit (LEO) satellites act as servers aggregating models from Internet of Remote Things (IoRT) clients, and it

TABLE I: Our work compared with previous works in aspects of FL and satellite security.

Name	Topic	Novelty in Sat. Net. / FL Security	Novelty in Sat-FL Application
FLGuardian [42]	FL Security	Evaluates the model by layer-wise clustering, assigns weights for each layer, then averages the weighted models.	N/A
FLAME [22]	FL Security	Defends against backdoor attacks based on clustering, clipping, and adding noise.	N/A
FLTrust [24]	FL Security	Assumes the server has bootstrapped clean data and can train a benign model for reference.	N/A
BaFFLe [23]	FL Security	Lets clients cross-validate each other's models and server-aggregate them according to feedback.	N/A
Med/Trimmed-mean [25]	FL Security	Aggregates the median of coordinates as the output model (Med). / Excluding outliers by trimming extreme values and then computing the average (Trimmed-mean).	N/A
Krum [26]	FL Security	Takes the model with minimal distance from its neighbors among all models as the global model for the next iteration.	Not Applicable (N/A)
SatelliteFL[7]	Sat-FL App.	N/A	Designs scheduling and bandwidth allocation algorithms for faster satellite-based FL convergence.
Matthiesen et al. [17]	Sat-FL App.	N/A	Summarizes FL in satellite constellations based on communication patterns among satellites, between satellites and ground stations, and constellation types.
FedSpace[19]	Sat-FL App.	N/A	Designs an FL aggregation scheme based on communication patterns between satellites and ground stations.
Semi-FedDA [44]	FL for Sat. Sec.	Applies FL on LEO satellites and ground servers for timely building-damage assessments.	N/A
SatOver[13]	Sat. Net. Sec.	Reveals a man-in-the-middle attack on satellite–ground communication leveraging LTE/5G stack vulnerabilities.	N/A
Firefly[14]	Sat. Net. Sec.	Demonstrates a spoofing attack on satellite–ground communication, where an adversary modifies downlink images in Earth observation tasks.	N/A
Salkield et al. [15]	Sat. Net. Sec.	Showcases a low-cost spoofing attack on satellite downlinks.	N/A
DFL-IDS [43]	FL for Sat. Sec.	Develops an FL-based detection method to protect satellite networks against cyberattacks.	N/A
SFL-LEO [45]	Sec. for FL on Sat. Net.	Proposes a homomorphic encryption-based FL aggregation scheme to achieve private model aggregation for IoRT at satellites.	Protects the privacy of IoRT devices whose models are aggregated at LEO satellites.
STARFed (ours)	Sec. for FL on Sat. Net.	Develops a link-status-aware, robust aggregation framework for the ground server to aggregate models from satellites.	Enhances the robustness of satellite–ground FL applications.

focuses on the privacy of IoRT models rather than aggregation robustness. A comparison between prior works and ours across different aspects of federated learning and satellite security is summarized in Table I, with related topics surveyed in the remainder of this section.

A. Ground Assisted Satellite FL

The emergence of satellite-based computation has sparked significant interest in ground-assisted satellite FL, driven by satellites' ability to provide global connectivity and unique data collection capabilities. Research in this domain broadly falls into two categories: systems where satellites act as primary data collectors and training nodes [7, 17, 19, 20, 40], and those where satellites serve as relay nodes for data collected by remote ground devices [16, 18, 21].

One of the challenges that has been extensively studied in the first category is the resource utilization of satellite-ground links. Yang et al. [7] addressed bandwidth limitations and intermittent connectivity through progressive block-wise quantization, while So et al. [19] proposed dynamic scheduling mechanisms to balance satellite idleness and model staleness. Matthiesen et al. [17] provided a comprehensive analysis of satellite-ground communication patterns, demonstrating superior performance of inter-satellite links in maintaining near-persistent connections compared to direct ground communication. Our work also falls into this category, where satellites act as primary data generators. For instance, using onboard cameras to capture Earth images for land change [46], air quality monitoring [47], or maritime surveillance [48]. In the second category, where satellites collect data from ground sensors, our framework still applies, as satellites can train FL models on collected data and return them for aggregation. However, this setting is less aligned with the FL principle that data should remain at its source. Addressing how to privately offload ground data to satellites is beyond our scope.

System heterogeneity is the main concern in the second category. Razmi et al. [40] developed frameworks addressing both data non-IID and connection heterogeneity, later extending their work to leverage intra-orbit inter-satellite links for improved training robustness [20]. For remote area applications, several works [16, 18, 21] explored satellites as computational relays for resource-constrained ground devices, with Han et al.

[16] specifically focusing on accelerating FL convergence in such scenarios.

Despite these advances in addressing communication and heterogeneity challenges, how security defects of S-T links could affect FL robustness remains unexplored. This oversight is particularly critical as these links represent the most exposed and vulnerable component of space-ground FL systems. A compromised communication link can undermine the entire FL process, highlighting the importance of link security alongside other system optimizations.

B. ML for Satellite-based Task Security

Beyond optimizing FL for satellite communication, machine learning has also been applied to satellite tasks. Salim et al. [43] used FL to detect threats in harsh satellite environments, while [44] employed federated and semi-supervised techniques for privacy-preserving Earth observation. These works focus on securing satellite tasks, not on ML/FL security itself. The most relevant work to ours is SFL-LEO [45], which utilizes homomorphic encryption to safeguard the privacy of IoRT devices in LEO satellite networks. However, it assumes satellites aggregate models from IoRT devices and emphasizes privacy preservation rather than aggregation robustness.

C. Satellite Spoofing Attacks

S-T links suffer from spoofing and hijacking attacks stemming from physical constraints such as long transmission distances and predictable signal paths [49, 50]. While historically focused on Global Navigation Satellite System (GNSS) signal protection through signal processing [51] and more recent deep learning approaches [52], the threat landscape has expanded with the advent of software-defined radio (SDR) technology and increasingly complex S-T infrastructures.

Recent research has demonstrated the increasing accessibility and sophistication of satellite communication attacks. Li et al. [13] revealed how SDR-based false satellites can execute man-in-the-middle attacks against satellite networks and ground devices by exploiting protocol vulnerabilities. In the context of earth observation systems, Salkield et al. [14] demonstrated how low-cost radio equipment (under $1000) could manipulate satellite data downlinks to fabricate or mask environmental events in NASA's Fire Information for Resource Management System (FIRMS) [53]. Salkield et al. [15] showed that effective overshadowing attacks could be executed at distances up to 1km using modest hardware (∼$2000), affecting both legacy and modern satellite systems.

In the context of space-ground machine learning, these security vulnerabilities pose equivalent risks, where the adversary compromising S-T links could not only tamper with benign models but also inject poisoned models.

D. Poisoning Defenses for FL

Existing defenses against FL poisoning attacks can be broadly categorized into three approaches:

Clustering-based Aggregation: These methods use clustering to separate benign models from poisoned ones. FLAME [22] applies HDBSCAN [54] clustering with clipping and noise addition, while Krum [26] selects as the global model the one closest to a specified number of neighbors. Li et al. [55] recently used Euclidean distance to detect poisoned models and clipping to improve aggregation robustness as in [22], dubbed E&C method. Beyond that, the authors presented a privacy-preserving E&C to protect client privacy while ensuring the aggregation robustness. The main drawback of the clustering-based approach is the risk of including poisoned models in aggregation, especially when adversarial clients outnumber the benign clients — potentially leading to an aggregated model dominated by poisoned updates.

Statistical Aggregation: Building upon the simple parameter averaging of FedAvg [41], Trimmed-Mean [25] excludes extreme values before averaging, while Median [25] uses coordinate-wise median for aggregation. FLPurifier [56] decouples the client's model into an encoder and a classifier. The encoders from clients are averaged while the classifiers are weighted averaged based on their deviation degree from the average classifier of all clients. More recently, FLGuardian [57] performs poisoned model filtering by weighting different layers of a model, accounting for the fact that poisoning at different depths has varying impacts on model performance. However, these center-seeking methods are less effective for FL on non-IID data, where the unbalanced data results in the models not having a representative central reference.

Trust-based Validation: These approaches use either trusted data or client feedback for validation. FLTrust [24] relies on a clean root dataset, which is difficult to obtain in satellite-based FL due to bandwidth limitations and the satellite's diverse geographical coverage. BaFFLe [23]'s client-based validation fails with non-IID data since clients cannot effectively validate models trained on different data distributions.

Two key limitations of existing aggregations are their sensitivity to non-IID data (common in satellite-based FL as discussed in Section II-A) and robustness consistency against diverse attacks. Most defenses target specific attacks — Krum [26] focuses on Gaussian Byzantine attacks, while FLAME [22] and BaFFLe [23] address backdoor attacks. As shown in Section VII-B, no existing aggregation scheme is robust under all tested attacks with non-IID data.

III. Preliminaries

A. Federated Learning

Federated Learning (FL) is a distributed machine learning paradigm that enables multiple clients to collaboratively train a shared model without exposing their local training data. In each training epoch $t \in \{1, \ldots, T\}$, each selected client $i \in \{1, \ldots, n\}$ trains a local model w_i based on the previous global model G_{t-1} using its local data D_i, and sends it to the server for aggregation into a new global model G_t.

Several aggregation mechanisms have been proposed, with FedAvg being the most widely used. In FedAvg, the global model is updated by weighted averaging of local models: $G_t = \sum_{i=1}^{n} s_i \times \frac{w_i}{s}$, where $s_i = \|D_i\|$ and $s = \sum_{i=1}^{n} s_i$. However, malicious clients may falsify their dataset sizes to amplify the impact of their updates. Therefore, equal weights ($s_i = \frac{1}{n}$)

are more commonly employed in practice, resulting in $G_t = \sum_{i=1}^{n} \frac{w_i}{n}$.

B. Poisoning Attacks Against FL

Poisoning attacks aim to manipulate the training process to corrupt the resulting model. In FL, these attacks can be categorized into two types:

Data Poisoning: The adversary manipulates the training data of compromised clients by modifying, adding, or removing examples. For instance, in label-flipping attacks [58], labels of training examples are changed while keeping their features unchanged.

Model Poisoning: Rather than manipulating training data, the adversary directly modifies the parameters of the updated model and scales it up to maximize attack impact or down to evade detection.

IV. Problem Statement

In this section, we present our system model, which introduces the role of each entity involved in satellite-based FL. We then describe the threat model. Finally, we introduce the problem definition and our design goals.

A. System Model

Our system model, illustrated in Figure 1, comprises three components: satellites, ground relays, and a centralized server. This architecture facilitates distributed learning across satellites via flexible ground access points.

The server initiates the training process by distributing a global model G_t to satellites through ground relays for a specific task. Satellites, acting as FL clients, train models using data locally observed or collected from sensors at remote terrestrial areas. Upon completion, they return updated models $\{w_i^t\}_{i=1}^K$ to the server via ground relays, which may be different relays from those used for initial distribution. The server then aggregates these updates into a new global model G_{t+1} for the next training epoch, continuing until the termination criteria are met. Both server-relay and satellite-terrestrial communications occur through unicast channels.

Ground relays are terrestrial devices ranging from compact satellite phones to large ground stations, establishing direct satellite links. In addition to model forwarding, they extract S-T link measurements l_i^t during model reception and relay this information to the server alongside the models.

We also consider client dropout due to the limited communication window and unstable S-T links. For a total number of N satellites, K ($K < N$) clients are involved in the training in each epoch t.

B. Threat Model

Among entities, we assume that satellites are honest and secure due to their isolation and limited physical accessibility as discussed in Section I. Additionally, we assume the centralized server is secure since it can be protected with dedicated security resources as a critical single infrastructure point.

The vulnerability lies in the S-T links for model transmission, which are susceptible to over-the-air (OTA) Man-in-the-Middle (MitM) adversaries. These adversaries possess varying capabilities and can launch different types of poisoning attacks against the FL training process. We classify adversaries based on their radio capabilities and the degree of S-T link control they can achieve:

Channel Interference Adversary: This represents the most basic form of attack, where an OTA MitM can interfere with the downlink model transmission. The adversary injects noise into the updated models through parameter-flipping (PF) attacks [59], replacing a fraction of model parameters with random values. This attack requires no knowledge about the model architecture or training data, aligning with the limited capabilities of channel interference adversaries.

Eavesdrop and Overshadow Adversary: This more sophisticated adversary can eavesdrop on the uplink S-T channel to obtain the global model G_t and overshadow the downlink S-T channel during model updates. The adversary launches the Model Poisoning Attack based on Fake client (MPAF) [60] that generates a poisoned model $w_i^{t\prime}$ by subtracting a Gaussian noise from the global model and then applying a factor λ to enlarge or reduce the poisoned model's effectiveness. The flexibility in choosing λ makes MPAF attacks more challenging to defend and mitigate.

Task-Aware Adversary: A task-aware adversary combines the radio capabilities of eavesdropping and overshadowing adversaries with knowledge of the FL task and training data structure. Specifically, under this model, we assume an adversary can eavesdrop on uplink channels during model distribution, compromise confidentiality, and infer the task or data type by launching membership inference attacks [61] or data reconstruction attacks [62] against the models. Leveraging this knowledge, the adversary can forge poisoned data to retrain the intercepted model and inject it by hijacking the downlink channel [14, 15] when the satellite transmits its honestly trained models. We consider three types of data poisoning attacks:

- Untargeted label-flipping (ULF) attack [58]: For a classification task with C classes, the adversary transforms labels y to $(y \bmod C) + 1$, aiming to degrade the overall model performance.
- Targeted label-flipping (TLF) attack [58]: The adversary maps all training labels to a single target class y_t, forcing the poisoned model to misclassify any input as the chosen class.
- Backdoor attack [63]: The adversary injects a trigger pattern into training samples x to create poisoned samples while changing their labels to a target class y_t. This creates a hidden backdoor in which the poisoned model classifies data with the trigger as y_t while behaving normally on clean data.

Data poisoning attacks are harder to detect than model poisoning attacks as they produce poisoned models that more closely resemble benign ones. However, they require a longer attack window for training on poisoned data. We note that data poisoning attacks cannot be launched by overshadowing the uplink channel, as the brief transmission window precludes

TABLE II: Poisoning attacks according to radio capabilities. Abbreviations: Atk. - Attack, M/DP - Model/Data Poisoning, UL - Uplink, DL - Downlink, PD - Poisoned Data.

Adversary Capability	Attack	Atk. Type	Atk. Pattern
Channel Interference	PF	MP	Interfere UL/DL
Eavesdrop & Overshadow	MPAF	MP	Eavesdrop on UL; Overshadow DL
Task-Aware	ULF TLF Backdoor	DP	Eavesdrop on UL; Train on PD; Overshadow DL

eavesdropping, training, and overshadowing within a single communication cycle. Table II summarizes the OTA MitM adversary capabilities and associated attacks.

Beyond OTA MitM adversaries, we consider that ground relays may be malicious. A malicious relay can execute any of the aforementioned poisoning attacks and can also fabricate S-T link information sent to the server. Furthermore, we assume relays are aware of the server's model aggregation scheme and can adapt their attack strategies accordingly. The unsecure communication paths are highlighted in red in Figure 1. Note that adversaries can only corrupt benign models but cannot legitimize poisoned ones.

C. Problem Definition and Design Goals

Given our system and threat model, we formally define the problem as follows: Consider an S-T FL system where K out of N benign satellite clients exchange models with the server via K out of H ground relays in each epoch. Among these relays, M are malicious and report forged link information, while P models are poisoned by either OTA MitM adversaries or malicious relays.

In each training epoch t, the server receives a set of K model-link information pairs:

$$\mathcal{L}_t = \{(w_i^t, l_i^t)\}_{i=1}^K \tag{1}$$

where w_i^t represents the model update forwarded by relay i and l_i^t denotes the corresponding S-T link information. Within this set $\mathcal{L}_t$, B pairs contain benign models and legitimate link information, while O pairs contain poisoned models and forged link information. The relationship between these parameters is illustrated by an example with $K = 10$ in Figure 2, where legitimate information is denoted in green and malicious information is in red.

Based on the received untrusted tuple set $\mathcal{L}_t$ at epoch t, our work aims to design a framework that achieves:

- *Robustness*: The training process should maintain resilience against all attacks summarized in Table II.
- *Accuracy*: The final global model G_T after T training epochs should achieve higher accuracy compared to existing defense mechanisms.
- *Privacy*: The link information l_i^t reported by relay i should preserve relay privacy by being less identifiable than the corresponding model update w_i^t. Formally, we evaluate it with the distinctiveness of the link information and models, calculated as the mean pairwise Euclidean distance between the normalized samples.

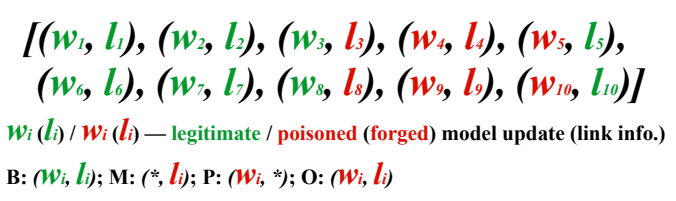


Fig. 2: Illustration of $K = 10$ model-link pairs with adversarial patterns: benign pairs ($B = 4$), poisoned models ($P = 4$), forged link information ($M = 4$), and both poisoned models and forged link information ($O = 2$).

- *Efficiency*: The framework should introduce minimal communication overhead.

V. STARFed Design

Algorithm 1 outlines STARFed's procedure. In each epoch, the server receives model-link information pairs from relays (line 5) and authenticates link information (line 7), which is detailed in Section V-A. Section V-B presents how authenticated link information is leveraged for poison model detection using the hybrid link-model characteristic clustering approach (line 8). The server uses the reputation-based progressive filter, described in Section V-C, to trace relays' historical behaviors and ensures aggregation robustness against adaptive adversaries (lines 9-10). Finally, Section V-D describes how the models that pass both filters are aggregated into the global model (line 11).

Algorithm 1 STARFed

1: **Input:** H, $\mathcal{R}_0$, G_0, T ▷ H is the number of ground relays; $\mathcal{R}_0$ are relays' initial reputations; G_0 is the initial global model; T is the number of training epochs
2: **Output:** $\mathcal{R}_T$, G_T ▷ $\mathcal{R}_T$ are relays' reputations after T epochs; G_T is the global model after T epochs
3: **for** each training epoch t in $[1, T]$ **do**
4: **for** each relay i in $[1, K]$ **do** ▷ $K = (1-\rho)H$ is the number of active relays, where ρ is the dropout portion
5: $\mathcal{L}_{t-1} \leftarrow$ RelayReport(G_{t-1}) ▷ Server distributes G_{t-1} and receives model-link information pairs defined in (1)
6: **end for**
7: $\mathcal{I}_{t-1} \leftarrow$ CrowdLinkAuth$(l_1^{t-1}, \ldots, l_K^{t-1})$ ▷ $\mathcal{I}_{t-1} = \{\texttt{ind}_i^{t-1}\}_{i=1}^K$ where $\texttt{ind}_i^{t-1}$ is an S-T link status indicator
8: $\mathcal{W}_C^{t-1} \leftarrow$ LinkClusterFilter$(\mathcal{W}_{t-1}, \mathcal{I}_{t-1})$ ▷ $\mathcal{W}_{t-1} = \{w_i^{t-1}\}_{i=1}^K$ are all received models; $\mathcal{W}_C^{t-1}$ are models passed link-clustering filter
9: $\mathcal{W}_R^{t-1} \leftarrow$ RepProgFilter$(\mathcal{W}_{t-1}, \mathcal{I}_{t-1}, \mathcal{R}_{t-1})$ ▷ $\mathcal{W}_R^{t-1}$ are models passed reputation-based progressive filter
10: $\mathcal{R}_t \leftarrow$ UpdateReputation$(\mathcal{R}_{t-1}, \mathcal{I}_{t-1}, \mathcal{W}_C^{t-1})$
11: $G_t \leftarrow$ Aggragate$(\mathcal{W}_C^{t-1}, \mathcal{W}_R^{t-1})$
12: **end for**

A. Crowdsourced Link Authentication

The link information is defined as a tuple:

$$l = (\texttt{clean\_flag}, \texttt{DP}(\texttt{link\_samples})) \tag{2}$$

where $\texttt{clean\_flag} \in \{0, 1\}$ represents the relay's local assessment of link legitimacy (1 for clean, 0 for spoofed), `link_samples` contains raw physical link measurements, and $\texttt{DP}(\cdot)$ applies differential privacy to the measurements.

The server authenticates links through crowdsourced spoofing detection. Specifically, each relay contributes a data point for the spoofing detector where $\texttt{DP}(\texttt{link\_samples})$ is the data and `clean_flag` is the locally assessed label. STARFed does not specify the spoofing detection method used by ground relays since recent advances in this task [64, 65] can be implemented with sufficient precision while remaining lightweight. In these works, pretrained ML models are adapted for spoofing detection by modifying their final layers to adapt satellite I/O features. This approach requires no training data, is computationally lightweight, and achieves high accuracy (0.8–1) by aggregating packet-level predictions into reliable stream-level decisions.

At the server side, STARFed adopts an SVM-based classifier [66] that is continuously trained on link samples received over epochs through an online learning scheme. The online learning procedure is presented in Algorithm 2. As shown in line 5 of the algorithm, the server uses the classifier to predict spoofing status of link samples. Since the prediction results may be affected by the relays' dishonesty or environmental variants, the server's prediction may not agree with the relay's reported label. The online learning is a conservative approach where the classifier is only updated based on consensus predictions, as shown in lines 6 and 7. Besides, the server computes a link indicator $\texttt{ind} = \frac{\texttt{clean\_flag}+\texttt{server\_det}}{2}$ for each model update, where $\texttt{ind} \in \{0, 0.5, 1\}$ indicates agreement on spoofed (0) or clean (1) links, or disagreement (0.5) between relay and server assessments.

For the link samples, existing works have shown that a broad range of S-T link physical layer features such as Doppler shifts [67, 68], the direction of arrival (DoA) [69, 70], and signal quality monitoring (SQM) [71, 72] are effective for spoofing detection in various scenarios. Thus, STARFed does not specify features in `link_samples`. Generally, each `link_samples` sequence is defined as: $\texttt{link\_samples} = \{x_i\}_{i=1}^{m}$, where x_i represents the feature vector of a single sample and m denotes the sequence length.

To preserve relay privacy, we apply the Laplace mechanism [73]:

$$\texttt{DP}(x_i) = x_i + \texttt{Laplace}(\mu, \frac{1}{\epsilon}) \tag{3}$$

where μ is the expectation of the Laplace distribution and ϵ controls the privacy budget. The motivation of $\texttt{DP}(\cdot)$ is to ensure that the link samples used by the framework do not introduce more identifiable information compared with the aggregation scheme that does not use the crowdsourced link authentication. In short, we add randomness to the link information sent to the server to achieve the privacy guarantee defined in Section IV-C. More detailed analysis is introduced in Section VI-D.

Algorithm 2 CROWDLINKAUTH's Online Learning Procedure

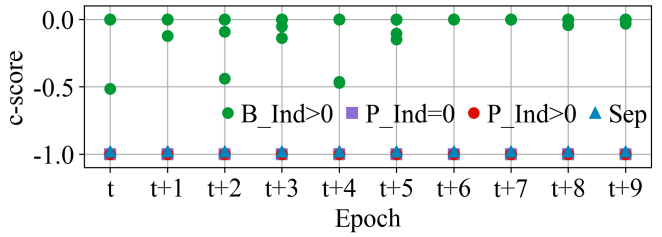
1: **Input:** M_t, α, γ ▷ M_t is the SVM-based classifier at epoch t; α is the learning rate; γ is the decay factor where $\gamma \in (0, 1)$
2: **Output:** M_{t+1} ▷ M_{t+1} is the classifier at epoch $t+1$
3: $\Delta M'_t \leftarrow 0$ ▷ Initialize a temporary classifier
4: **for** l_i in $\{l_i^t\}_{i=1}^{K}$ **do** ▷ l_i as defined in (2)
5: $\texttt{srv\_det}_i \leftarrow$ DETECT(M_t, $\texttt{DP}(\texttt{link\_samples}_i)$) ▷ $\texttt{srv\_det}_i$ is server's detection (0: spoofed, 1: clean)
6: **if** $\texttt{srv\_det}_i = \texttt{clean\_flag}_i$ **then**
7: $\Delta M'_t \leftarrow \Delta M'_t +$ COMPUTEUPDATE(M_t, l_i, α) ▷ Leverage l_i to update the classifier with α
8: **end if**
9: **end for**
10: $M_{t+1} \leftarrow \gamma \cdot M_t + (1-\gamma) \cdot \Delta M'_t$ ▷ Apply weighted update

B. Hybrid Link-Model Characteristic Clustering Filter

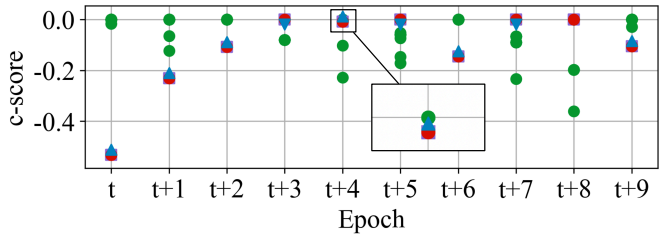
The hybrid link-model characteristic clustering filter (link-clustering filter) integrates both model clustering analysis and link indicator to detect poisoned updates. Following Nguyen et al. [22], we employ pairwise cosine distances between received models as the metric and cluster the models using the HDBSCAN algorithm [54]. The cosine distance effectively captures angular deviations between models while remaining invariant to scaling attacks, where adversaries attempt to evade detection by scaling poisoned models.

HDBSCAN dynamically determines cluster cardinality, identifying the majority cluster ($\texttt{cluster\_label} = 0$) and marking divergent models as outliers ($\texttt{cluster\_label} = -1$). While effective for IID scenarios where benign models are similar, this binary classification presents two limitations: *Non-IID Issue:* In non-IID settings, benign models inherently exhibit greater variance due to data heterogeneity. HDBSCAN's majority-based clustering identifies only the subset of similar models as the majority cluster, while marking other benign but diverse models as outliers. This degrades the FL performance by excluding valid training contributions from the global model updates. *Majority Attack Vulnerability:* More threateningly, coordinated attacks can exploit this majority-based clustering approach: When multiple adversaries align their poisoned models toward a common malicious objective (e.g., by uniformly flipping labels to a single target class as in TLF attacks), these poisoned models are similar and can outnumber the diverse benign models. Consequently, the poisoned cluster is identified as the majority, allowing the attack to successfully evade the defense.

To address these limitations, we first revise the binary clustering results (i.e. majority vs. outliers) as a continuous cluster score $\texttt{c-score} = \texttt{cluster\_label} \cdot \texttt{outlier\_score}$ that maps model evaluations to $(-1, 0]$, where $\texttt{outlier\_score} \in [0, 1)$ quantifies model isolation relative to the distribution. This evaluation maintains the



(a) MPAF Attack



(b) Backdoor Attack

Fig. 3: Model separation results based on `c-score` and `ind`.

majority cluster at 0 while distributing outliers across $(0, -1)$, enabling more nuanced filtering when combined with link indicators.

Figure 3 illustrates the `c-scores` of the benign models (in green) and models poisoned (in red and purple) by MPAF (3a) and backdoor (3b) attacks of 10 epochs of our non-IID data experiments which will be detailed in Section VII. As shown in Figure 3a, although benign models are clustered at 0 while poisoned ones are clustered at -1, many benign models are isolated from the majority and spread from (0,-1). In this case, these models will be excluded from aggregation by clustering-based filtering that only accepts the majority, validating the *non-IID issue*. Moreover, Figure 3b validates the *majority attack vulnerability*: The poisoned models are clustered as the majority and located at 0 in the plot at epochs t+3, 5, 7 and 8. In those epochs, the poisoned models outnumber the benign ones and will be accepted by the existing clustering-based approaches, potentially failing the overall training process, as we will show in Section VII.

Motivated by the observations, we leverage link indicators as an additional factor to separate benign models from poisoned ones as clustering alone is insufficient. Algorithm 3 details our separation mechanism that operates in two stages. First, the algorithm partitions models based on their link indicators, creating two sets: suspicious models with `ind` = 0 and non-suspicious models with `ind` $>$ 0 (lines 3-4). For the suspicious set, which likely contains poisoned models, we compute separation thresholds using the statistical properties of their `c-scores`: `up_sep` = max($\mathcal{S}$) + Var($\mathcal{S}$) (upper threshold, line 7) and `low_sep` = min($\mathcal{S}$) - Var($\mathcal{S}$) (lower threshold, line 8). These thresholds then separate non-suspicious models into sets $\mathcal{N}_{above}$ and $\mathcal{N}_{below}$ (lines 9-10). Models are accepted if either set contains at least φ proportion of the non-suspicious models (lines 12-17).

The rationale for the algorithm stems from the novelty

Algorithm 3 LINKCLUSTERFILTER's Separation Component

1: **Input:** $\mathcal{W}_t$, $\mathcal{C}_t$, $\mathcal{I}_t$, φ ▷ $\mathcal{C}_t$ are models' `c-scores`; φ is the separation factor
2: **Output:** $\mathcal{W}_C^t$
3: $\mathcal{S} \leftarrow \{\texttt{c-score}_i \mid \texttt{ind}_i = 0\}$ ▷ $\mathcal{S}$ are `c-scores` of models with suspicious link indicators
4: $\mathcal{N} \leftarrow \{\texttt{c-score}_i \mid \texttt{ind}_i \geq 0\}$ ▷ $\mathcal{N}$ are `c-scores` of models with non-suspicious indicators
5: **if** $\mathcal{S} = \emptyset$ or $\mathcal{N} = \emptyset$ **then return** $\emptyset$
6: **end if**
7: `up_sep` $\leftarrow$ max($\mathcal{S}$) + Var($\mathcal{S}$) ▷ `up_sep` is the upper separator based on the maximum and variance of $\mathcal{S}$
8: `low_sep` $\leftarrow$ min($\mathcal{S}$) + Var($\mathcal{S}$) ▷ `low_sep` is the lower separator based on the minimum and variance of $\mathcal{S}$
9: $\mathcal{N}_{above} \leftarrow \{w_i \mid \texttt{c-score}_i \in \mathcal{N}$ & $\texttt{c-score}_i >$ `up_sep`$\}$ ▷ $\mathcal{N}_{above}$ are non-suspicious models above `up_sep`
10: $\mathcal{N}_{below} \leftarrow \{w_i \mid \texttt{c-score}_i \in \mathcal{N}$ & $\texttt{c-score}_i <$ `low_sep`$\}$ ▷ $\mathcal{N}_{below}$ are non-suspicious models below `low_sep`
11: $\mathcal{W}_C^t \leftarrow \emptyset$
12: **if** $|\mathcal{N}_{above}| \geq \varphi \cdot |\mathcal{N}|$ **then**
13: $\quad \mathcal{W}_C^t \leftarrow \mathcal{N}_{above}$
14: **end if**
15: **if** $|\mathcal{N}_{below}| \geq \varphi \cdot |\mathcal{N}|$ **then**
16: $\quad \mathcal{W}_C^t \leftarrow W_{Ct} \cup \mathcal{N}_{below}$
17: **end if**
18: **return** $\mathcal{W}_C^t$

of evaluating models along two complementary dimensions: model similarity and communication reliability. The first dimension, quantified by the `c-score`, measures how close a model is to the distribution, capturing deviations that may indicate poisoning. However, in non-IID settings, benign models may diverge, making similarity alone insufficient. The second dimension, indicated by the link status `ind`, reflects whether the model was transmitted through a trustworthy channel. This provides an orthogonal trust source: adversarial updates may appear statistically consistent yet still arrive via compromised links. A benign model must be validated on both dimensions before being accepted.

For the suspicious set (models with `ind`=0), we further refine the separation using thresholds derived from the statistical properties of their `c-score`. Specifically, `up_sep` and `low_sep` extend the boundaries around the most extreme suspicious scores. Anchoring at max($\mathcal{S}$) + Var($\mathcal{S}$) and min($\mathcal{S}$) - Var($\mathcal{S}$) ensures that no poisoned models are mistakenly accepted, while the variance term adaptively adjusts for dispersion. When suspicious models are tightly clustered (low variance), the thresholds remain narrow, excluding similar poisoned models. When suspicious models are diverse (high variance), the thresholds expand to tolerate such diversity.

As shown in Figure 3, the algorithm effectively handles multiple scenarios: The upper (and lower) separators for each epoch are shown as the blue triangles pointing up (and down). Models with link indicators greater than 0 are marked circles,

and the ones with link indicators equal to 0 are marked rectangles. The legend shows benign (B) and poisoned (P) models with their indicator values ($B\_Ind > 0$, $P\_Ind = 0$, and $P\_Ind > 0$) and the upper and lower separators (Sep). We discuss the separation approach based on the following representative cases:

1) Under MPAF attacks (Figure 3a), poisoned models form outlier clusters while many benign ones disperse across $(0, -1)$. The separation effectively captures these scattered models, addressing the *non-IID issue*. Note that if a binary model evaluation such as the one in [22] is applied, only models evaluated at `c-score` = 0 will be aggregated and all benign models evaluated in $(0, -1)$ are missed. This comparison shows STARFed's inclusiveness of benign models in non-IID data scenarios.
2) During backdoor attacks (Figure 3b), when poisoned models infiltrate the majority cluster (epochs t+3, 5, 7), the separation still accepts benign models with $\texttt{ind} > 0$, even when they are in the minority. This case demonstrates how the framework avoids the *majority attack vulnerability*.
3) More importantly, even when an aggressive malicious relay attempts to deceive the server by reporting $\texttt{ind} > 0$ for poisoned models (shown as red circles in the plots), these models are filtered out as their `c-scores` align more closely with other poisoned models rather than the benign ones.
4) Conversely, if a malicious relay reports $\texttt{ind} = 0$ for benign models, this may affect the separation thresholds through max($\mathcal{S}$), min($\mathcal{S}$), and Var($\mathcal{S}$). However, as discussed in Section IV-B, a malicious relay cannot control a benign model's position in the `c-score` spectrum as it cannot fabricate a benign model, preventing arbitrary manipulation of the separation thresholds.

While the separation provides robust filtering in most cases, corner cases exist (e.g., epoch t+8 in Figure 3b) where no models meet the acceptance criteria. We address these limitations through the reputation-based progressive filter, which is detailed in the following section.

C. Reputation-based Progressive Filter

Before presenting the reputation-based filter, we introduce STARFed's reputation system that evaluates relay reliability. Following Algorithm 1 (line 10), the server updates relay reputations after each training epoch based on their link indicator values and the link-clustering filter results. The reputation update for relay i at epoch t is defined as:

$$\texttt{rep\_update}_i = \begin{cases} -\rho & \text{if } \texttt{ind}_i < 1 \\ \rho & \text{if } \texttt{ind}_i = 1 \text{ and } w_i \in \mathcal{W}_C \\ -c\rho & \text{if } \texttt{ind}_i = 1 \text{ and } w_i \notin \mathcal{W}_C \end{cases} \quad (4)$$

where $\rho > 0$ is the reputation factor and $c > 1$ is a weighting factor for severe penalties.

The reputation system handles six distinct scenarios based on the link indicator value and model acceptance status:

- *Suspicious Link, Accepted Model* ($\texttt{ind} = 0$, $w_i \in \mathcal{W}_C$): Indicates a disturbing S-T link condition where both relay and detector fail to identify a clean channel. While this incurs a moderate penalty $-\rho$, it's unlikely to be caused by a malicious relay, as intentionally reporting $\texttt{ind} = 0$ for an acceptable model would only damage the relay's reputation without providing any poisoning effects.
- *Suspicious Link, Rejected Model* ($\texttt{ind} = 0$, $w_i \notin \mathcal{W}_C$): The system penalizes the relay with $-\rho$, regardless of whether it's legitimate but under attack or malicious.
- *Uncertain Link, Accepted Model* ($\texttt{ind} = 0.5$, $w_i \in \mathcal{W}_C$): Suggests either imperfect link assessment or variation from other relays' samples. Results in a moderate penalty $-\gamma$.
- *Uncertain Link, Rejected Model* ($\texttt{ind} = 0.5$, $w_i \notin \mathcal{W}_C$): Similar to case 3, receiving penalty $-\gamma$ discourages inconsistent reporting.
- *Clean Link, Accepted Model* ($\texttt{ind} = 1$, $w_i \in \mathcal{W}_C$): The ideal case where the link assessments match the link-clustering result. The relay receives a reward γ.
- *Clean Link, Rejected Model* ($\texttt{ind} = 1$, $w_i \notin \mathcal{W}_C$): The most severe case, receiving an increased penalty $-c\gamma$. This strict penalty applies to: (1) malicious relays attempting to inject poisoned models with forged link information, and (2) benign relays potentially under sophisticated OTA MitM attacks that can manipulate both the physical link characteristics and model content, making even honest relays report clean links for compromised models.

The reputation system assigns penalties ($-\rho$) for suspicious/uncertain links, rewards (ρ) for clean links with accepted models, and increased penalties ($-c\rho$) for clean links with rejected models.

Building on this reputation mechanism, the reputation-based filtering design should address two key requirements unique to satellite-based FL systems: 1) *Resilience to Intermittent Connectivity:* Due to frequent satellite dropouts, the filter must accommodate benign models from benign relays even after periods of disconnection. 2) *Recovery from Attacks:* Benign relays temporarily compromised by OTA MitM attacks should regain system trust once the attack ceases. To meet these principles, we define the link reputation filter that accepts models satisfying:

$$w_i \in \mathcal{W}_R^t \iff (\texttt{ind}_i > 0) \wedge (r_i^t > \texttt{rep\_thr}_t) \quad (5)$$

where the reputation threshold evolves as:

$$\texttt{rep\_thr}_t = \omega\rho t + r^0 \quad (6)$$

Here, r_i^t represents relay i's reputation at epoch t, $\omega \in (0, 1)$ is the progression factor, and r^0 denotes the initial reputation assigned to all relays.

The threshold increases linearly by $\omega\rho$ each epoch, where $\omega < 1$ ensures gradual progression. Specifically, a relay with reputation exactly at threshold ($r_i^t = \texttt{rep\_thr}_t$) can still contribute benign models for q epochs, provided $q < \frac{1}{\omega}$. Besides, even if a relay's reputation falls below the threshold, this design still allows legitimate relays to maintain participation and rewards consistent, honest behaviors, enabling eventual recovery.

D. Model Aggregation

STARFed's aggregation strategy balances *inclusiveness* and *selectivity*. Inclusiveness captures diverse training data from benign models, while selectivity filters out divergent poisoned models to prevent training failure. Specifically, it adapts based on the link-clustering filter's output. When the filter accepts more than half of the received models ($|\mathcal{W}_C^t| > \frac{|\mathcal{W}_t|}{2}$), indicating strong separation from the potentially malicious `c-score` range, we accept $\mathcal{W}_C^t$ only without considering $\mathcal{W}_R^t$ accepted by the reputation-based filter. This scenario satisfies both inclusiveness and selectivity requirements through reliable model separation. Conversely, when insufficient models are identified by clustering ($|\mathcal{W}_C^t| \leq \frac{|\mathcal{W}_t|}{2}$), including cases like epoch t+8 in Figure 3b where $\mathcal{W}_C^t = \emptyset$, we take the union of two filters' outputs ($\mathcal{W}_C^t \cup \mathcal{W}_R^t$) as finally accepted models. This approach is safe because weak separation indicates poisoned and benign models are not significantly distinguished in the `c-score` spectrum, making reputation-based decisions more reliable.

The final aggregation averages [41] the accepted models as the updated global model to maximize the inclusiveness. If both filters accept no model, STARFed computes the coordinate-wise median of model parameters [25] to maintain selectivity against extreme poisoned values as a fallback for extreme corner cases.

VI. Security Analysis

A. Robustness of the Link-Clustering Filter

We define the conditions that must be satisfied for a malicious relay to inject poisoned models into the system. We then analyze STARFed's robustness against such attacks based on the definitions.

Definition 1: (Separable) Let $\mathcal{W}$ be the set of all received models in an epoch. Define $\mathcal{W}_S \subset \mathcal{W}$ as the set of suspicious models with link indicator $\texttt{ind} = 0$, and $\mathcal{W}_N \subset \mathcal{W}$ as the set of non-suspicious models with link indicator $\texttt{ind} > 0$. Let $f : \mathcal{W} \to [0, 1)$ be the function computing the `c-score`. The set of separated models $\mathcal{W}_A$ is defined as:

$$\begin{aligned}\mathcal{W}_A = \{w_i \mid w_i \in \mathcal{W}_N \wedge \\ (f(w_i) < \texttt{low\_sep} \vee f(w_i) > \texttt{up\_sep})\}\end{aligned} \tag{7}$$

where `low_sep` and `up_sep` are the separation boundaries calculated according to Algorithm 3. The set of models is considered separable if:

$$\frac{|\mathcal{W}_A|}{|\mathcal{W}_N|} > \varphi \tag{8}$$

where φ is the separation factor.

Definition 2: (Evading) Given a set of separable models, a poisoned model w_p is evading if:

$$w_p \in \mathcal{W}_A \tag{9}$$

Based on Definitions 1 and 2, injecting a poisoned model requires satisfying two conditions simultaneously: (1) the set of models must be separable, and (2) the injected model must be evading. Consider the following cases:

If all adversaries forge link information with $\texttt{ind} > 0$ for their poisoned models, the link-clustering filter remains inactive due to lack of suspicious models, violating the separability condition in Definition 1. If all adversaries forge link information with $\texttt{ind} = 0$ for their poisoned models, these models form the suspicious set $\mathcal{W}_S$. This prevents any poisoned model from being evaded according to Definition 2, as $\mathcal{W}_A$ only contains models from $\mathcal{W}_N$.

Therefore, adversaries must adopt a mixed strategy where some relays must report $\texttt{ind} = 0$ to establish separability while others report $\texttt{ind} > 0$ to enable model evasion.

To execute this strategy successfully, relays reporting $\texttt{ind} = 0$ must carefully craft their poisoned models to form a suspicious range that excludes at least φ proportion of non-suspicious models, requiring knowledge of both φ and all other models' `c-scores`. The presence of even a *single* honest relay can disrupt this strategy. When an honest relay detects an OTA MitM attack and reports $\texttt{ind} = 0$, it expands the suspicious range. This expansion violates the guarantee of successful poisoned model injection. Moreover, such failed injection attempts lead to severe reputation penalties according to (4). Therefore, the successful injection of poisoned models requires adversary's *control over all relays*.

B. Robustness of Reputation-based Model Filter

We consider two scenarios that may affect the robustness of the reputation-based model filter. In the first scenario, the adversary is adaptive and changes its attack probability each round, trying to inject poisoned models while maintaining a high reputation for the relays it controls. In the second scenario, we consider that network irregularities and unpredictable environmental effects may lead honest relays to output noisy or inconsistent measurements, degrading their reputation.

For adaptive adversary, we let the attack probability be p_a at each round. The expected number of epochs before detection is $\frac{1}{p_a}$. However, the reputation penalty for a detected attack ($-c\rho$) can be set to exceed potential gains from reputation building (ρ). Specifically, for an OTA adversary that cannot forge the link measurement to pass STARFed's link authentication, its expected reputation gain is $E[rep_{OTA}] = p_a \times (-\rho) + (1 - p_a) \times \rho = (1 - 2p_a) \times \rho$. In this case, the adaptive adversary has to make its overall attack probability p_a lower than half to maintain a positive reputation, enforcing it to contribute more benign models than poisoned ones.

For an adversary that can control ground relays to send forged link measurements, the expected reputation for a malicious ground relay is $E[rep_{MGR}] = p_a \times (-c\rho) + (1 - p_a) \times \rho = (1 - (1 + c) \times p_a) \times \rho$. Since $c > 1$, the adversary has to make $p_a < \frac{1}{1+C} < \frac{1}{2}$ to maintain a positive reputation. The flexibility of tuning the reputation penalty factor c enforces an adaptive and stricter model contribution requirement compared to OTA adversaries.

Following the above discussion, an honest ground relay, under unpredictable and irregular S-T link conditions, can achieve a positive reputation as long as the probability of irregularity (p_{ir}) is lower than half (the same requirement as p_a). If a ground relay is suffering from network irregularity with a probability higher than half, the reputation-based filter

should exclude its contributing model from aggregation for the system's overall robustness, regardless of whether it is benign.

C. Integrated Analysis for Overall Robustness

The above analysis only considers cases where an OTA adversary hijacking the S-T link can never pass the link authentication, and the model from a malicious ground relay can always be excluded by the link-clustering filter. Now, we integrate STARFed's components with errors and analyze its overall robustness.

Assuming the SMV for link authentication has error rate E_{auth} and the hybrid link-clustering filter has error rate E_{hyb}. The condition for a poison model evading the hybrid detector is discussed in Section VI-A. We now analyze the conditions under which an adversary can inject a (link-clustering filter-bypassing) poisoned model with a positive reputation. Combining the analysis in Section VI-B with components' error rates, the expected reputation for a malicious OTA adversary is $E[rep^{err}_{OTA}] = p_a \times (1 - E_{auth}) \times (-\rho) + p_a \times E_{auth} \times (1 - E_{hyb}) \times (-c\rho) + p_a \times E_{auth} \times E_{hyb} \times \rho + (1 - p_a) \times \rho$. To maintain the overall positive reputation, the adversary has to keep its attack probability $p_a < \frac{1}{2-((1+c)E_{hyb}+(1-c))E_{auth}}$. The boundary is approximately $\frac{1}{2}$ relaxed by an error factor $((1 + c)E_{hyb} + (1 - c))E_{auth}$ in the dominator.

For an adversary controlling a malicious ground relay, its expected reputation under component error is $E[rep^{err}_{MGR}] = p_a \times (1 - E_{hyb}) \times (-c\rho) + p_a \times E_{hyb} \times \rho + (1 - p_a) \times \rho$. To maintain the overall positive reputation, the adversary has to keep its attack probability $p_a < \frac{1}{(c+1)(1-E_{hyb})}$. Given the reputation penalty factor $c > 1$, the boundary for malicious ground really under error is also approximately $\frac{1}{2}$, relaxed by an error factor $1 - E_{hyb}$.

The analysis for honest ground relays under unpredictable S-T link conditions is similar. Based on the above discussion, we conclude that even in the scenarios in which link authentication and hybrid link-clustering filter misclassify link measurements and models with specific error rates, the framework still forces the adversary to contribute more benign models and poisoned ones with a small error factor determined by the error rates of individual components.

D. Privacy Analysis

We now analyze how applying differential privacy (DP) to link status samples reduces relays' privacy exposure. Recall that we measure privacy exposure by the distinctiveness of a distribution in Section IV-C. Formally, given a model or link status vector v_i, we normalize it as $u_i = \frac{v_i}{\|v_i\|_2}$, and compute the pairwise Euclidean distance as $\sqrt{2\big(1 - \delta_{ij}\big)}$, where $\delta_{ij} = u_i^\top u_j$. Thus, distinctiveness of the distribution is defined as

$$\text{Disc} = \frac{2}{n(n-1)} \sum_{i<j} \sqrt{2\big(1 - \delta_{ij}\big)} \tag{10}$$

Adding Laplace noise with scale $\frac{1}{\epsilon}$ (as in Eq. (3)) makes the normalized vectors noisier. In the case where link statuses are highly distinct (i.e., the average cosine similarity δ_{ij} is negative), decreasing ϵ increases δ_{ij} toward zero. This, in turn, reduces the overall distinctiveness. In practice, we tune ϵ such that $\text{Disc}_{\text{link}} < \text{Disc}_{\text{model}}$, which ensures that link status information is less distinctive (and thus reveals less privacy) compared to FL models.

VII. Evaluation

Section VII-A outlines our experimental setup. We compare STARFed's effectiveness against state-of-the-art FL aggregation schemes across multiple datasets in Section VII-B. The effectiveness of the framework's key components is examined in Section VII-C. Section VII-D reveals the influence of non-IID degrees, client dropout rates, and adversarial ratios. The system's communication and privacy overhead are measured and discussed in Section VII-E.

All experiments run on a machine equipped with an Intel Core i9-12900k CPU (3.2 GHz), 32 GB RAM, and an NVIDIA RTX 3090 GPU. All the attacks, defenses and model training are implemented in Python using PyTorch [74] and Torchvision [75] libraries. The implementation code will be made publicly available upon publication of this paper.

A. Experimental Settings

FL Datasets and models. We evaluate STARFed on three benchmark datasets. MNIST [76] contains 70,000 grayscale handwritten digit images (28×28 pixels). CIFAR-10 [77] consists of 60,000 color images (32×32 pixels) spanning 10 classes. EuroSAT [78] comprises 27,000 Sentinel-2 satellite images (64×64 pixels) with 10 land use classes, which is particularly suitable for satellite-based FL scenarios.

We implement different model architectures tailored to each dataset's unique sizes and characteristics. For the MNIST experiments, we employ two architectures: a basic convolutional neural network (CNN) [76] structured with 2 convolutional and 2 fully connected layers, and a multilayer perception (MLP) [79] designed with 3 fully connected layers. CIFAR-10 tasks utilize a ResNet50 architecture [80]. The EuroSAT dataset is trained using ResNet18 [80] and a lightweight variant of EfficientNet [81]. Table III summarizes the dataset splits and corresponding model parameters.

S-T link spoofing and detection. Due to the absence of publicly available general-purpose S-T link spoofing datasets, we evaluate STARFed's spoofing detection capabilities using a benchmark dataset for GNSS spoofing research called the Texas Spoofing Test Battery dataset (TEXBAT) [82]. It is a widely used dataset in GNSS spoofing research that was developed at the University of Texas at Austin's Radionavigation Laboratory. It was specifically designed to provide researchers with real-world GNSS spoofing scenarios collected from physical experiments for testing and evaluating spoofing detection algorithms. The dataset includes recordings of GPS signals subjected to various spoofing attacks under controlled laboratory conditions and has become the most popular and commonly used benchmark in GNSS spoofing detection research [71, 72]. Our experiments specifically use `dataset3` (spoofed signals with static, low-power advantage scenarios) and `cleanStatic` (clean signals in static scenarios) recordings as spoofing and non-spoofing samples, respectively.

TABLE III: Datasets and models used in our evaluations.

Datasets	#Training	#Testing	Model	#params
MNIST	60k	10k	CNN MLP	∼582k ∼199k
CIFAR-10	50k	10k	ResNet50	∼26M
EuroSAT	21.6k	5.4k	ResNet18 EfficientNet-Light	∼196k ∼2M

As discussed in Section V-A, the framework can accommodate any domain-specific S-T link features. Without loss of generality, we extract six S-T link characteristics: carrier-to-noise ratio (C/N_0), I-Q samples, and correlator outputs (early/prompt/late) from GNSS signal tracking. The feature extraction is motivated by state-of-the-art GNSS spoofing detection works [66, 67, 71, 72] in which the features are shown to be representative and effective for spoofing detection. While the latter three features are GNSS-specific, this specialization does not affect STARFed's general design principles. Although our experiments use GNSS spoofing data, the physical-layer observables we extract—SNR proxies (C/N_0), complex baseband I/Q statistics, and matched-filter outputs near the timing estimate (early/prompt/late)—are generic to satellite–terrestrial receivers. Spoofing in any S–T system disturbs these quantities due to attacker imperfections in power control, synchronization, and phase alignment. We thus view GNSS as a publicly available instance of the broader radio frequency (RF)-layer detection problem. In non–spread-spectrum systems (e.g., general S–T links), correlator taps are replaced by preamble/pilot matched-filter taps. These yield similar timing-related structures—such as peak symmetry and slope—that enable spoofing detection. Despite differing signal primitives, both GNSS and non-spread-spectrum systems produce structurally similar receiver outputs, allowing spoofing detection methods based on peak perturbation to generalize across protocols.

Each relay forwards 10 (i.e., $m = 10$ in `link_samples` definition) feature vectors to the server, where the centralized spoofing detector processes individual samples. The final spoofing detection employs majority voting across 10 feature vectors. The progression factor ω and reputation reward ρ for the reputation-based filter (6) are set 0.001 and 0.01, respectively. We note that developing a complete state-of-the-art spoofing detector for general-purpose S-T links extends beyond our current scope.

B. Comparison Results

Existing Defenses. We compare STARFed with the basic aggregation scheme FedAvg [41] and state-of-the-art defense mechanisms (Krum [26], Median [25], Trimmed-Mean [25], FLAME [22], and FLGuardian [57]) introduced in Section II-D.

Additionally, we implement a link-aware baseline that combines the link filter with FLGuardian. This baseline only accepts models with `ind` $= 1$, then aggregates them with FLGuardian.

Adversarial Settings. We configure a network of $N = 50$ satellite clients. To create non-IID conditions, we first sort each dataset by labels and then partition it into shards of equal size. Each client randomly receives a fixed number of shards. For the MNIST training set (60k samples), we create shards of 240 samples, each with identical labels. Each client receives $\frac{60k \text{ samples}}{240 \text{ samples} \times 50 \text{clients}} = 5$ shards, ensuring at most 5 different label types for each client. We apply similar partitioning to CIFAR-10 and EuroSAT datasets.

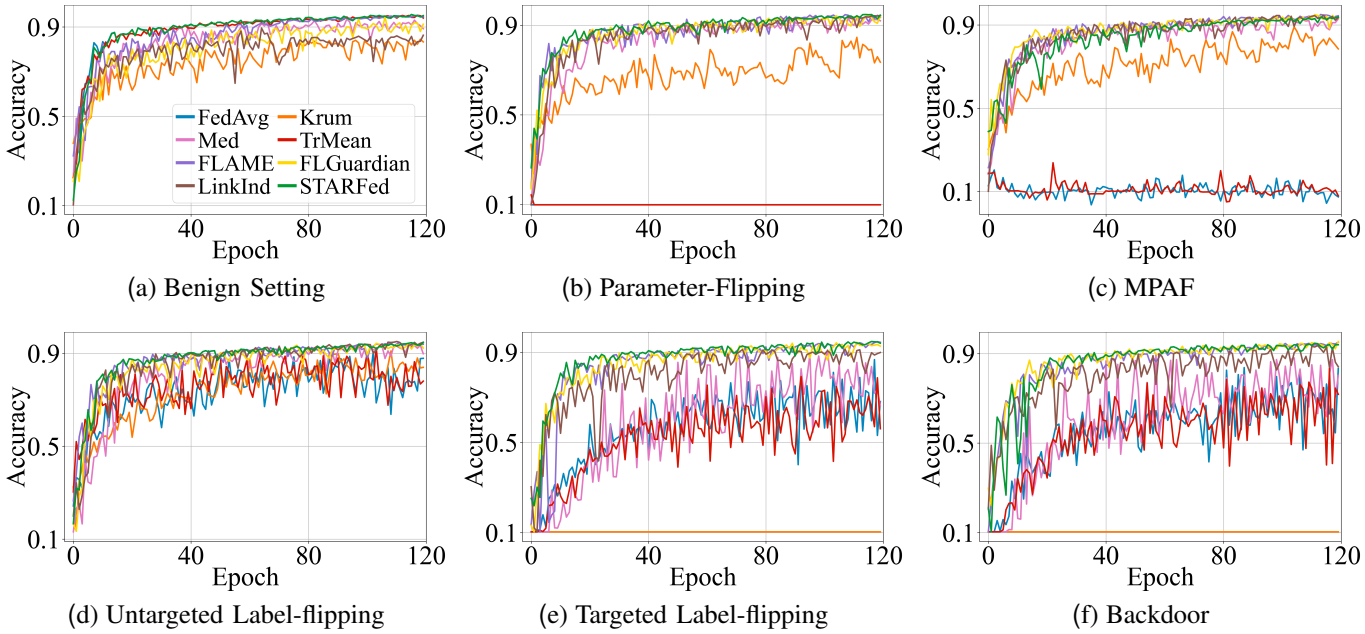
Following our threat model depicted in Figure 2, we test with $K = 20$ satellite clients per global training epoch, each communicating with the server through a unique ground relay. Each relay i forwards a model (w_i) and reports associated link information (l_i) to the server. The adversarial setup consists of $P = 7$ poisoned models (35%) injected by OTA MitM adversaries or malicious ground relays. Concurrently, $M = 7$ malicious ground relays report forged link information. Among these, $O = 4$ relays accompany poisoned models with dishonest link information, where two report inconsistent link information and two present consistent but contradictory information (indicating clean links for poisoned models). The remaining $(M - O = 3)$ relays report inconsistent link information for benign models. In total, $(M + P - O = 11)$ out of 20 (55%) model-link pairs contain malicious content, leaving $B = 9$ honest pairs with benign models and legitimate link information.

Defense Effectiveness. Figure 4 (for MNIST-CNN study) and Table IV (for other studies) present comprehensive evaluation results across different datasets and attack scenarios. Bold numbers in Table IV indicate the best performance in each setting. When STARFed achieves the best performance, ↑ shows its improvement over the second-best defense; when another method performs better, ↓ indicates STARFed's gap from the best performance. From the plots and the table, we can see that STARFed is the *only* robust scheme against *all* types of attacks on evaluated datasets, while maintaining high accuracies in benign settings. In the EuroSAT-ResNet18 study under the backdoor attack, it outperforms the best link-unaware aggregation (FLGuardian) by 15.6%. While in the worst case, it only falls behind the best defense by 5.5% as under the untargeted label-flipping attack in the same study.

While FLAME matches STARFed's performance in the MNIST-CNN study, it shows vulnerability in more challenging scenarios: targeted label-flipping and backdoor attacks in CIFAR-10 and the two EuroSAT studies, and untargeted label-flipping attacks in CIFAR-10 and EuroSAT-EfficientNet studies. This degradation occurs because non-IID data distribution in complex tasks increases model diversity, making it harder to distinguish between benign and poisoned models as detailed in Section V-B.

Moreover, the clustering-based approaches (i.e., Krum across all studies and FLAME in EuroSAT-ResNet18) achieve only ∼10% accuracy under targeted label-flipping and backdoor attacks. This again validates the majority attack vulnerability discussed earlier in the paper (Section V-B) that these methods can mistakenly aggregate poisoned models when they form the majority, causing all predictions to follow the adversary's targeted class.

Besides, we notice that the link-aware baseline (LinkInd) stands under various attacks, however, it's accuracies drop $\sim 10\%$ compares with STARFed. This is due to its conserva-



(a) Benign Setting (b) Parameter-Flipping (c) MPAF

(d) Untargeted Label-flipping (e) Targeted Label-flipping (f) Backdoor

Fig. 4: Comparative results on MNIST dataset using CNN.

TABLE IV: Overall Comparison Results

Def.\Atk.	Benign	PF	MPAF	ULF	TLF	Backdoor	Benign	PF	MPAF	ULF	TLF	Backdoor
	MNIST - MLP						CIFAR-10 - ResNet50					
FedAvg	89.9	10.5	8.5	63.2	66.7	87.8	64.7	13.1	10.0	50.4	26.9	30.4
Krum	85.6	87.0	84.8	83.8	10.3	10.3	39.0	37.6	52.8	6.3	10.0	10.0
Med	84.1	80.5	77.0	71.2	54.2	67.5	63.4	64.6	10.0	48.3	10.0	10.0
TrMean	90.3	10.6	10.7	79.5	70.6	88.3	65.1	3.7	10.0	47.8	17.8	19.9
FLAME	86.1	81.8	**89.2**	89.4	88.7	89.6	66.7	64.0	65.1	44.0	24.2	18.0
FLGuardian	84.4	**90.4**	88.7	89.1	87.9	90.7	62.2	**66.5**	**66.5**	40.5	64.3	**65.5**
LinkInd	83.7	**90.4**	89.1	90.1	87.2	81.3	63.2	64.7	65.2	62.7	**65.1**	63.6
STARFed	**90.7(↑0.4)**	89.9(↓0.5)	88.3(↓0.9)	**90.2(↑0.1)**	**90.5(↑1.8)**	**91.1(↑1.5)**	**68.3(↑3.6)**	65.5(↓1)	64.6(↓1.9)	**64.2(↑1.5)**	64.1(↓1)	63.8(↓1.7)
	EuroSAT - ResNet18						EuroSAT - EfficientNet-Light					
FedAvg	72.9	24.9	11.6	46.6	16.9	29.2	64.8	13.6	10.3	53.8	33.2	29.9
Krum	49.1	56.0	51.2	58.5	10.4	10.5	47.9	50.4	35.9	23.2	11.2	10.4
Med	57.2	62.2	13.2	46.9	23.0	10.8	61.9	61.4	11.0	51.7	18.6	30.9
TrMean	65.1	22.2	11.5	46.4	18.6	18.2	67.0	21.3	11.9	49.1	21.9	36.4
FLAME	69.8	71.0	69.2	73.0	10.8	11.2	67.5	64.3	65.5	51.4	20.3	29.5
FLGuardian	67.2	67.6	**72.5**	**74.6**	**71.9**	57.4	59.6	63.7	71.4	53.0	65.5	**67.5**
LinkInd	63.2	65.4	68.0	65.2	65.1	**74.1**	66.8	65.7	**68.3**	64.9	66.1	63.6
STARFed	**74.1(↑1.2)**	**72.2(↑1.2)**	67.4(↓5.1)	69.1(↓5.5)	68.9(↓3)	73.0(↓1.1)	**70.0(↑2.5)**	**68.3(↑2.6)**	64.2(↓4.1)	**69.3(↑4.4)**	**67.2(↑1.1)**	65.5(↓2)

tive strategy that discards models with $\texttt{ind} < 1$, potentially missing benign ones. We emphasize that our primary objective is to develop a robust defense against attacks rather than surpassing standard FL performance benchmarks.

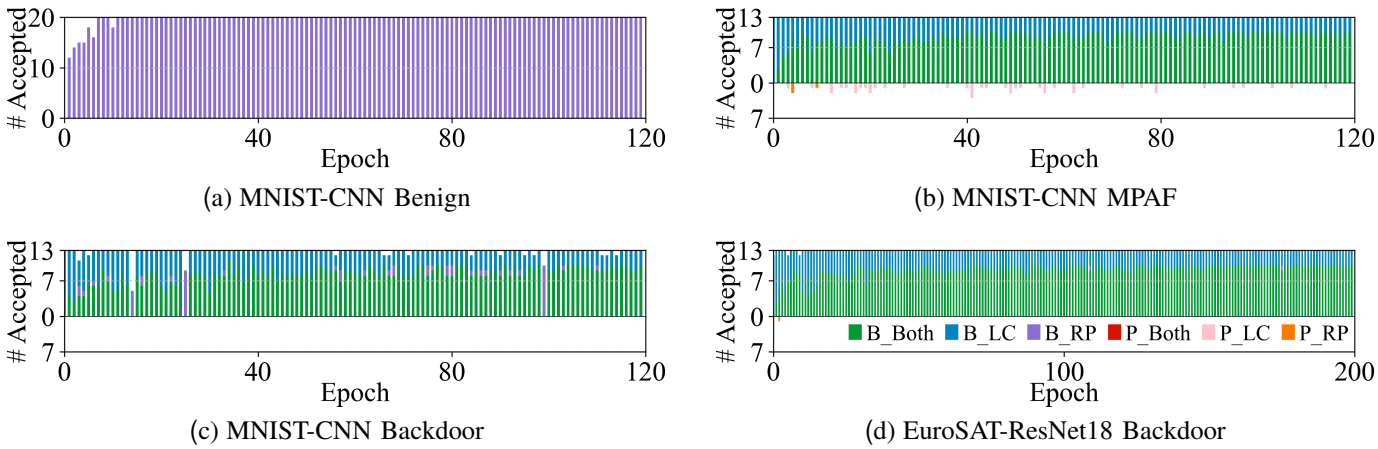
C. Component Evaluation

We analyze STARFed's filtering components through the MNIST-CNN study across three representative scenarios: benign setting, MPAF, and backdoor attacks. Figure 5 visualizes the performance of both filters, where counts above the x-axis represent accepted benign models and below represent accepted poisoned models. The legend uses prefixes 'B_' for benign and 'P_' for poisoned models, with suffixes indicating acceptance by: 'Both_' (both filters), 'LC_' (link-clustering filter only), or 'RP_' (reputation-based progressive filter only).

In the benign setting (Figure 5a), with no link indicators showing potential attacks, the link-clustering filter remains inactive. The reputation-based progressive filter initially accepts fewer models by missing clients with a reputation below the threshold due to dropout but stabilizes after 20 epochs to consistently accept all 20 models, demonstrating its resilience to intermittent connectivity described in Section V-C.

Under MPAF attacks (Figure 5b), where adversaries add noise to corrupt models, the link-clustering filter successfully identifies and accepts all 13 benign models while rejecting most poisoned ones. Besides, the reputation-based filter maintains steady acceptance of approximately 10 models per iteration, with most of these models (shown in green) being accepted by both filters, indicating the agreement between the two filters.

Similarity for backdoor attacks (Figure 5c), both filters effectively separate benign from poisoned models. This is particularly evident in the EuroSAT-ResNet18 case (Figure 5d), where STARFed maintains robust performance while other clustering-based defenses (Krum and FLAME) fail to detect the attack, as shown in Table IV. This matches our design idea: when models are similar, such as all models are benign, the framework relies less on clustering since the separation



(a) MNIST-CNN Benign (b) MNIST-CNN MPAF

(c) MNIST-CNN Backdoor (d) EuroSAT-ResNet18 Backdoor

Fig. 5: Benign and poisoned models accepted by STARFed's two filters.

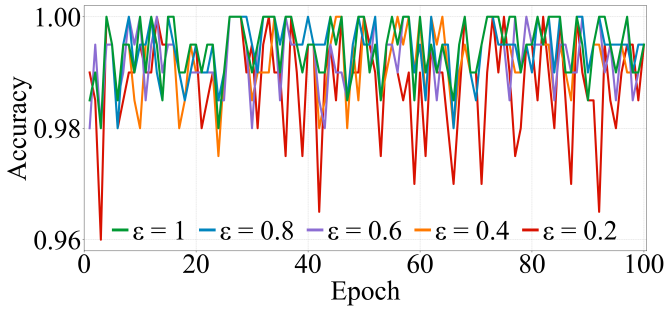
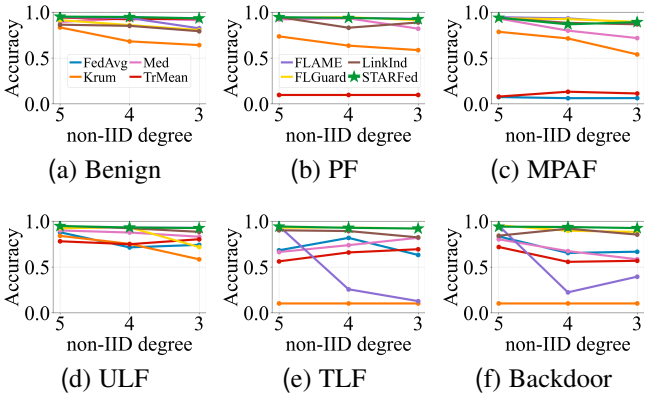


Fig. 6: Crowdsourcing spoofing detector accuracies with various ϵ.



(a) Benign (b) PF (c) MPAF

(d) ULF (e) TLF (f) Backdoor

Fig. 7: Last epoch accuracies of compared defenses in various non-IID settings under different attacks.

threshold becomes less effective in this case and relies more on relay's reputation. When models are diverse, the clustering filter becomes dominant in the decision process, while the reputation penalty factor and progression factor keep tracking relay's behaviors. In iterations where the models are not diverse, the reputation system takes over and filter models, as shown in the sporadic purple bars in Figure 5c.

We next evaluate the effectiveness of STARFed's crowdsourced link authentication module under different privacy constraints. Figure 6 shows the centralized spoofing detector's accuracy in classifying reported link samples across 100 epochs, with varying differential privacy budgets (ϵ). The detector maintains robust performance with high privacy protection: accuracy consistently exceeds 98% for ϵ values of 1.0, 0.8, and 0.6. Even under stricter privacy settings with ϵ reduced to 0.4 and 0.2, the detector's accuracy remains above 96%, demonstrating the module's resilience to privacy-preserving noise.

D. Impact of Training Heterogeneity and Adversarial Settings

We evaluate STARFed's robustness under varying degrees of data heterogeneity and adversarial presence using the MNIST-CNN study.

Impact of non-IID degree. We investigate STARFed's performance under increasing non-IID data distribution by adjusting the shard size from 240 (baseline) to 300 and 400, restricting each client to receive at most 5, 4, or 3 different labels respectively. A smaller number of unique labels for each client indicates a higher degree of non-IID distribution. As shown in Figure 7, STARFed (green) maintains consistent performance across all non-IID settings under various attacks. While other defenses experience performance degradation with increasing non-IID degrees, STARFed demonstrates stable performance across all attack scenarios.

Impact of dropout and adversarial intensity. Based on the parameters introduced in Section IV-C, we define the number of corrupt relays as the ones sending either poisoned models or forged link status, minus the ones sending both. Thus, the adversarial ratio is computed as $\frac{(P+M-O)}{K}$. We examine STARFed's resilience by increasing both dropout rates and adversary proportions from the baseline (P-M-O/K = 7-7-4/20, 50% adversaries) to medium (9-9-6/20, 60%) and high (11-11-8/20, 70%) intensity. Figure 8 shows that under baseline and medium adversarial intensities, STARFed (green) maintains robust performance across all attack types. Under high intensity, STARFed demonstrates varied resilience. It maintains stable performance against parameter-flipping, targeted label-flipping, and backdoor attacks (Figures 8a, 8d,

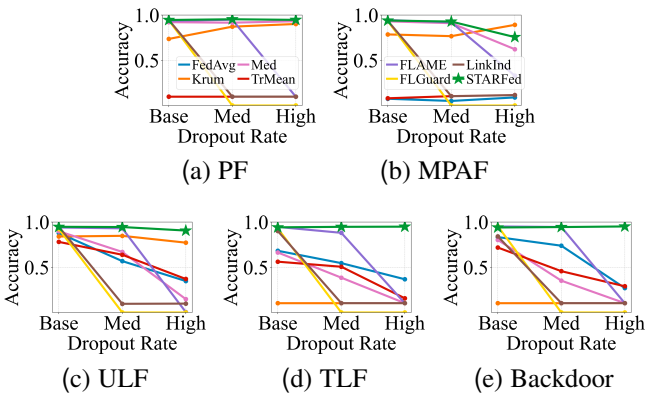


Fig. 8: Last epoch accuracies of compared defenses in various dropout and adversarial settings under different attacks.

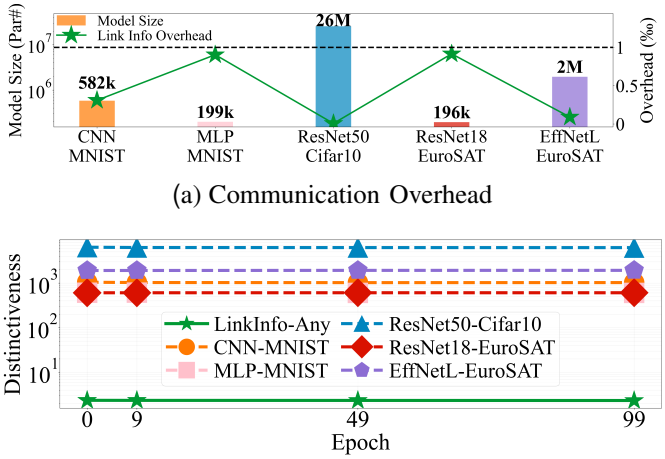


Fig. 9: Communication and privacy overhead of link information compared with model updates.

and 8e) while showing some degradation in MPAF and untargeted label-flipping scenarios (Figures 8b and 8c). However, it still achieves the best or second-best performance among all defenses. FLAME, the only defense comparable to STARFed under baseline and medium settings, exhibits significant performance deterioration under high adversarial intensity.

E. Overhead Evaluation

We next evaluate the communication and privacy overhead of STARFed.

Communication Overhead. STARFed integrates S-T link information transmission with model updates, requiring no additional communication rounds between relays and the server. The only overhead comes from transmitting link information consisting of m S-T link feature vectors as lists of float numbers and one binary flag (`clean_flag`) per model. Following the overhead evaluation method applied in communication-efficient FL research [83, 84], Figure 9a compares this overhead across different settings, with bars showing model sizes in the number of parameters on a logarithmic scale (left y-axis) and stars indicating the link information to model size ratio in percentage (right y-axis). The results demonstrate that STARFed achieves enhanced robustness with minimal communication overhead—less than 1‰ across all model architectures.

Privacy Analysis. We evaluate privacy overhead using distinctiveness as a metric, calculated as the mean pairwise Euclidean distance between normalized samples. For both link information and models, we first perform normalization: link samples across features and models coordinate-wise. Higher distinctiveness indicates greater potential for sample identification, thus higher privacy risk. Figure 9b presents these results on a logarithmic scale, with link information distinctiveness (shown as stars) collected from MNIST-CNN serving as a representative case. The distinctiveness correlates with content size—larger models (shown with different colors and markers) exhibit higher distinctiveness by orders of magnitude. Link information is significantly smaller than model parameters, introducing minimal additional privacy risk compared to the inherent exposure from model transmission.

VIII. Conclusion

This paper presents STARFed, a novel framework that rethinks FL security in satellite-based contexts by focusing on the unique vulnerabilities of S-T communication links. Unlike existing FL systems that assume compromised clients, STARFed addresses the more realistic threat model where launched satellites remain secure while S-T links become the primary vulnerability. STARFed uses (1) a crowdsourced link authentication system that leverages physical characteristics from multiple ground relays to detect spoofing attacks, (2) a hybrid link-clustering model filter that identifies poisoned models through both link indicators and model characteristic analysis and (3) a reputation-based progressive filter that imposes penalties on malicious relays attempting to inject poisoned models while rewarding honest behavior, effectively deterring sustained attacks. Our experimental results demonstrate that STARFed significantly enhances accuracies in practical satellite-based training scenarios while introducing minimal overhead. The framework's ability to distinguish between benign and poisoned models, even in the presence of sophisticated poisoning attacks and malicious ground relays, represents a significant advance in secure satellite-based FL.

References

[1] SpaceX. (2024) Transporter-11 Mission. Accessed: 2024-10-23. [Online]. Available: https://www.spacex.com/launches/mission/?missionId=transporter11

[2] T. Pultarova. (2024) SpaceX to launch 1st space-hardened NVIDIA AI GPU on upcoming rideshare mission. Accessed: 2024-10-23. [Online]. Available: https://www.space.com/ai-nvidia-gpu-spacex-launch-transporter-11

[3] Y. Michalevsky and Y. Winetraub, "Spacetee: Secure and tamper-proof computing in space using cubesats," *arXiv preprint arXiv:1710.01430*, 2017.

[4] M. Sabt, M. Achemlal, and A. Bouabdallah, "Trusted execution environment: What it is, and what it is not," in *2015 IEEE Trustcom/BigDataSE/Ispa*, vol. 1. IEEE, 2015, pp. 57–64.

[5] L. Marelli and G. Testa, "Scrutinizing the eu general data protection regulation," *Science*, vol. 360, no. 6388, pp. 496–498, 2018.

[6] C. Amirfar. (2023) Remote Sensing from Space: What Norms Govern? Accessed: 2024-10-23. [Online]. Available: https://www.justsecurity.org/86114/remote-sensing-from-space-what-norms-govern/

[7] C. Yang, J. Yuan, Y. Wu, Q. Sun, A. Zhou, S. Wang, and M. Xu, "Communication-efficient satellite-ground federated learning through progressive weight quantization," *IEEE Transactions on Mobile Computing*, 2024.

[8] J.-P. A. Yaacoub, H. N. Noura, and O. Salman, "Security of federated learning with iot systems: Issues, limitations, challenges, and solutions," *Internet of Things and Cyber-Physical Systems*, vol. 3, pp. 155–179, 2023.

[9] P. Rieger, T. Krauß, M. Miettinen, A. Dmitrienko, and A.-R. Sadeghi, "Crowdguard: Federated backdoor detection in federated learning," *arXiv preprint arXiv:2210.07714*, 2022.

[10] X. Liu, T. Kim, and D. E. Shasha, "Bounce: A high performance satellite-based blockchain system," *Network*, vol. 5, no. 2, p. 9, 2025.

[11] S.-K. Liao, W.-Q. Cai, W.-Y. Liu, L. Zhang, Y. Li, J.-G. Ren, J. Yin, Q. Shen, Y. Cao, Z.-P. Li *et al.*, "Satellite-to-ground quantum key distribution," *Nature*, vol. 549, no. 7670, pp. 43–47, 2017.

[12] A. Reezwana, T. Islam, X. Bai, C. F. Wildfeuer, A. Ling, and J. A. Grieve, "A quantum random number generator on a nanosatellite in low earth orbit," *Communications Physics*, vol. 5, no. 1, p. 314, 2022.

[13] W. Li, Y. Li, H. Li, Y. Chen, Y. Wang, J. Lan, J. Wu, Q. Wu, J. Liu, and Z. Lai, "The dark side of scale: Insecurity of direct-to-cell satellite mega-constellations," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 2024, pp. 149–149.

[14] E. Salkield, S. Birnbach, S. Kohler, R. Baker, M. Strohmeier, and I. Martinovic, "Firefly: spoofing earth observation satellite data through radio overshadowing," 2023.

[15] E. Salkield, M. Szakály, J. Smailes, S. Köhler, S. Birnbach, M. Strohmeier, and I. Martinovic, "Satellite spoofing from a to z: on the requirements of satellite downlink overshadowing attacks," in *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2023, pp. 341–352.

[16] D.-J. Han, S. Hosseinalipour, D. J. Love, M. Chiang, and C. G. Brinton, "Cooperative federated learning over ground-to-satellite integrated networks: Joint local computation and data offloading," *IEEE Journal on Selected Areas in Communications*, 2024.

[17] B. Matthiesen, N. Razmi, I. Leyva-Mayorga, A. Dekorsy, and P. Popovski, "Federated learning in satellite constellations," *IEEE Network*, 2023.

[18] T. K. Rodrigues and N. Kato, "Hybrid centralized and distributed learning for mec-equipped satellite 6g networks," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 4, pp. 1201–1211, 2023.

[19] J. So, K. Hsieh, B. Arzani, S. Noghabi, S. Avestimehr, and R. Chandra, "Fedspace: An efficient federated learning framework at satellites and ground stations," *arXiv preprint arXiv:2202.01267*, 2022.

[20] N. Razmi, B. Matthiesen, A. Dekorsy, and P. Popovski, "On-board federated learning for dense leo constellations," in *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022, pp. 4715–4720.

[21] H. Chen, M. Xiao, and Z. Pang, "Satellite-based computing networks with federated learning," *IEEE Wireless Communications*, vol. 29, no. 1, pp. 78–84, 2022.

[22] T. D. Nguyen, P. Rieger, R. De Viti, H. Chen, B. B. Brandenburg, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen *et al.*, "{FLAME}: Taming backdoors in federated learning," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1415–1432.

[23] S. Andreina, G. A. Marson, H. Möllering, and G. Karame, "Baffle: Backdoor detection via feedback-based federated learning," in *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2021, pp. 852–863.

[24] X. Cao, M. Fang, J. Liu, and N. Z. Gong, "Fltrust: Byzantine-robust federated learning via trust bootstrapping," in *Network and Distributed Systems Security Symposium*. NDSS, 2021.

[25] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *International conference on machine learning*. Pmlr, 2018, pp. 5650–5659.

[26] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Advances in neural information processing systems*, vol. 30, 2017.

[27] 3rd Generation Partnership Project (3GPP), "NR; NR and NG-RAN Overall Description; Stage-2," 3GPP, Tech. Rep. TS 38.300, 2022, release 17. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/138300_138399/138300/17.00.00_60/

[28] J. Krause, "Non-terrestrial networks (ntn)," 3rd Generation Partnership Project (3GPP) website, 2024, published May 14, 2024; updated July 4, 2025. [Online]. Available: https://www.3gpp.org/technologies/ntn-overview

[29] Apple, "Use Emergency SOS via satellite on your iPhone," Apple Support, Feb. 2025, accessed 2025-08-24. [Online]. Available: https://support.apple.com/en-us/101573

[30] ATT, "AT&T and AST SpaceMobile Take Connectivity to New Heights," AT&T News, Feb. 2025, accessed 2025-08-24. [Online]. Available: https://about.att.com/story/2025/ast-spacemobile-video-call.html

[31] Starlink, "Starlink Direct to Cell (Business)," Starlink website, 2025, accessed 2025-08-24. [Online]. Available: https://www.starlink.com/us/business/direct-to-cell

[32] T-Mobile, "Direct to Cell Satellite Phone Service

(T-Satellite with Starlink)," T-Mobile website, 2025, accessed 2025-08-24. [Online]. Available: https://www.t-mobile.com/coverage/satellite-phone-service

[33] M. Jaffar and N. Chuberre, "NTN & Satellite in Rel-17 & 18," 3rd Generation Partnership Project (3GPP) Partner News, Jul. 2022, article first published in HIGHLIGHTS Issue 03, Oct. 2021. [Online]. Available: https://www.3gpp.org/news-events/partner-news/ntn-rel17

[34] S. Burleigh, K. Fall, V. Cerf, R. Durst, K. Scott, and H. Weiss, "Bundle protocol version 7," Internet Engineering Task Force (IETF), RFC 9171, Jan. 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9171

[35] D. J. Israel, J. P. Swinski, J. Wilmot, S. Strege, B. Anderson, P. Jain, and C. Matusow, "Implementing delay/disruption tolerant networking for nasa's plankton, aerosol, clouds, ocean ecosystem (pace) mission," in *16th International Conference on Space Operations (SpaceOps 2021)*. International Astronautical Federation (IAF), May 2021, paper No. SpaceOps-2020,1,1,3,x1326; published by NASA's Goddard Space Flight Center. [Online]. Available: https://ntrs.nasa.gov/api/citations/20210013734/downloads/DTN%20SpaceOps%20Paper%202021%20Israel%20D.pdf

[36] Starlink Insider, "Starlink Ground Station Locations (2025)," Starlink Insider website, 2025, published online, updated as of 2025; accessed 2025-08-24. [Online]. Available: https://starlinkinsider.com/starlink-gateway-locations/

[37] P. B. de Selding, "Eutelsat: 38 oneweb gateways in service, 4 more being built; oneweb gen 2 awaits eu decision on iris2 multi-orbit network," Space Intel Report, Oct. 2024, published online. [Online]. Available: https://www.spaceintelreport.com/eutelsat-38-oneweb-gateways-in-service-4-more-being-built-oneweb-gen-2-awaits-eu-decision-on-iris2-multi-orbit-network/

[38] P. Couturier, Y. Lagsir, and D. Georgiev, "Extending operational ground networks for in-orbit satellites," AWS Public Sector Blog, Aug. 2025, accessed 2025-08-24. [Online]. Available: https://aws.amazon.com/blogs/publicsector/extending-operational-ground-networks-for-in-orbit-satellites/

[39] Kongsberg Satellite Services (KSAT), "The KSAT Global Ground Station Network," KSAT website, 2025, accessed 2025-08-24. [Online]. Available: https://www.ksat.no/ground-network-services/the-ksat-global-ground-station-network/

[40] N. Razmi, B. Matthiesen, A. Dekorsy, and P. Popovski, "Ground-assisted federated learning in leo satellite constellations," *IEEE Wireless Communications Letters*, vol. 11, no. 4, pp. 717–721, 2022.

[41] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[42] X. Zhou, X. Chen, S. Liu, X. Fan, Q. Sun, L. Chen, M. Qiu, and T. Xiang, "Flguardian: Defending against model poisoning attacks via fine-grained detection in federated learning," *IEEE Transactions on Information Forensics and Security*, 2025.

[43] S. Salim, N. Moustafa, M. Hassanian, D. Ormod, and J. Slay, "Deep-federated-learning-based threat detection model for extreme satellite communications," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 3853–3867, 2023.

[44] Y. Zhang, Y. Gong, and Y. Guo, "Semi-supervised federated learning for assessing building damage from satellite imagery," in *ICC 2024-IEEE International Conference on Communications*. IEEE, 2024, pp. 3821–3826.

[45] S. S. Hassan, U. Majeed, Z. Han, and C. S. Hong, "Sfl-leo: Secure federated learning computation based on leo satellites for 6g non-terrestrial networks," in *NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2023, pp. 1–5.

[46] M. C. Hansen, P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland *et al.*, "High-resolution global maps of 21st-century forest cover change," *science*, vol. 342, no. 6160, pp. 850–853, 2013.

[47] A. Van Donkelaar, R. V. Martin, M. Brauer, R. Kahn, R. Levy, C. Verduzco, and P. J. Villeneuve, "Global estimates of ambient fine particulate matter concentrations from satellite-based aerosol optical depth: development and application," *Environmental health perspectives*, vol. 118, no. 6, pp. 847–855, 2010.

[48] Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "A sar dataset of ship detection for deep learning under complex backgrounds," *remote sensing*, vol. 11, no. 7, p. 765, 2019.

[49] C. Li, X. Sun, and Z. Zhang, "Effective methods and performance analysis of a satellite network security mechanism based on blockchain technology," *IEEE Access*, vol. 9, pp. 113 558–113 565, 2021.

[50] A. Diro, S. Kaisar, A. V. Vasilakos, A. Anwar, A. Nasirian, and G. Olani, "Anomaly detection for space information networks: A survey of challenges, techniques, and future directions," *Computers & Security*, vol. 139, p. 103705, 2024.

[51] H. Wen, P. Y.-R. Huang, J. Dyer, A. Archinal, and J. Fagan, "Countermeasures for gps signal spoofing," in *Proceedings of the 18th international technical meeting of the satellite division of the institute of navigation (ION GNSS 2005)*, 2005, pp. 1285–1290.

[52] P. Borhani-Darian, H. Li, P. Wu, and P. Closas, "Detecting gnss spoofing using deep learning," *EURASIP Journal on Advances in Signal Processing*, vol. 2024, no. 1, p. 14, 2024.

[53] NASA. (2024) NASA Fire Information for Resource Management System. Accessed: 2024-10-30. [Online]. Available: https://firms.modaps.eosdis.nasa.gov/

[54] R. J. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Pacific-Asia conference on knowledge discovery and data mining*. Springer, 2013, pp. 160–172.

[55] X. Li, X. Yang, Z. Zhou, and R. Lu, "Efficiently achiev-

ing privacy preservation and poisoning attack resistance in federated learning," *IEEE Transactions on Information Forensics and Security*, 2024.

[56] J. Zhang, C. Zhu, X. Sun, C. Ge, B. Chen, W. Susilo, and S. Yu, "Flpurifier: backdoor defense in federated learning via decoupled contrastive training," *IEEE Transactions on Information Forensics and Security*, 2024.

[57] X. Zhou, X. Chen, S. Liu, X. Fan, Q. Sun, L. Chen, M. Qiu, and T. Xiang, "Flguardian: Defending against model poisoning attacks via fine-grained detection in federated learning," *IEEE Trans. Inf. Forensics Secur.*, vol. 20, pp. 5396–5410, 2025. [Online]. Available: https://doi.org/10.1109/TIFS.2025.3570119

[58] 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, ESORICS 2020, Guildford, UK, September 14–18, 2020, Proceedings, Part I 25*. Springer, 2020, pp. 480–501.

[59] C. Qian, M. Zhang, Y. Nie, S. Lu, and H. Cao, "A survey of bit-flip attacks on deep neural network and corresponding defense methods," *Electronics*, vol. 12, no. 4, p. 853, 2023.

[60] X. Cao and N. Z. Gong, "Mpaf: Model poisoning attacks to federated learning based on fake clients," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3396–3404.

[61] 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)*. IEEE, 2017, pp. 3–18.

[62] J. C. Zhao, A. Sharma, A. R. Elkordy, Y. H. Ezzeldin, S. Avestimehr, and S. Bagchi, "Loki: Large-scale data reconstruction attack against federated learning through model manipulation," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024, pp. 1287–1305.

[63] T. Liu, Y. Zhang, Z. Feng, Z. Yang, C. Xu, D. Man, and W. Yang, "Beyond traditional threats: A persistent backdoor attack on federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 19, 2024, pp. 21 359–21 367.

[64] G. Oligeri, S. Sciancalepore, S. Raponi, and R. Di Pietro, "Past-ai: Physical-layer authentication of satellite transmitters via deep learning," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 274–289, 2022.

[65] M. Kang, S. Park, and Y. Lee, "A survey on satellite communication system security," *Sensors*, vol. 24, no. 9, p. 2897, 2024.

[66] X. Zhu, T. Hua, F. Yang, G. Tu, and X. Chen, "Global positioning system spoofing detection based on support vector machines," *IET Radar, Sonar & Navigation*, vol. 16, no. 2, pp. 224–237, 2022.

[67] A. Jovanovic, C. Botteron, and P.-A. Fariné, "Multitest detection and protection algorithm against spoofing attacks on gnss receivers," in *2014 IEEE/ION Position, Location and Navigation Symposium-PLANS 2014*. IEEE, 2014, pp. 1258–1271.

[68] A. Broumandan, A. Jafarnia-Jahromi, V. Dehghanian, J. Nielsen, and G. Lachapelle, "Gnss spoofing detection in handheld receivers based on signal spatial correlation," in *Proceedings of the 2012 IEEE/ION position, location and navigation symposium*. IEEE, 2012, pp. 479–487.

[69] M. L. Psiaki, T. E. Humphreys, and B. Stauffer, "Attackers can spoof navigation signals without our knowledge. here's how to fight back gps lies," *IEEE Spectrum*, vol. 53, no. 8, pp. 26–53, 2016.

[70] G. Xu, F. Shen, M. Amin, and C. Wang, "Doa classification and ccpm-pc based gnss spoofing detection technique," in *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. IEEE, 2018, pp. 389–396.

[71] K. Ali, E. G. Manfredini, and F. Dovis, "Vestigial signal defense through signal quality monitoring techniques based on joint use of two metrics," in *2014 IEEE/ION Position, Location and Navigation Symposium-PLANS 2014*. IEEE, 2014, pp. 1240–1247.

[72] E. G. Manfredini, F. Dovis, and B. Motella, "Validation of a signal quality monitoring technique over a set of spoofed scenarios," in *2014 7th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)*. IEEE, 2014, pp. 1–7.

[73] A. M. Shahmiri, C. W. Ling, and C. T. Li, "Communication-efficient laplace mechanism for differential privacy via random quantization," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 4550–4554.

[74] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[75] S. Marcel and Y. Rodriguez, "Torchvision the machine-vision package of torch," https://pypi.org/project/torchvision/, 2010, accessed: 2025-09-01.

[76] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[77] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep. 0, 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[78] P. Helber, B. Bischke, A. Dengel, and D. Borth, "Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 7, pp. 2217–2226, 2019.

[79] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[80] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual

learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[81] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[82] A. Lemmenes, P. Corbell, and S. Gunawardena, "Detailed analysis of the texbat datasets using a high fidelity software gps receiver," in *Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2016)*, 2016, pp. 3027–3032.

[83] Z. Huo, Y. Fan, and Y. Huang, "A communication-efficient federated text classification method based on parameter pruning," *Mathematics*, vol. 11, no. 13, p. 2804, 2023.

[84] Y. Dong, W. Hou, X. Chen, and S. Zeng, "Efficient and secure federated learning based on secret sharing and gradients selection," *J. Comput. Res. Dev*, vol. 57, pp. 2241–2250, 2020.

Zizheng Liu received the B.E. in Computer Science and Technology at Northeastern University, Shenyang, China, in 2018 and the M.S. in Computer Science at Columbia University in 2019. He is pursuing a Ph.D. with the Department of Computer Science at Purdue University. His research interests include security and privacy issues in distributed and mobile networks.

Bharat K. Bhargava (Life Fellow, IEEE) is a Professor at the Department of Computer Science at Purdue University. He is the Founder of the *IEEE Symposium on Reliable and Distributed Systems*, the *IEEE Conference on Digital Library*, and the *ACM Conference on Information and Knowledge Management*. He is the Editor-in-Chief of four journals and serves on over ten editorial boards of international journals. He has published hundreds of research articles and has won five best paper awards in addition to the Technical Achievement Award and Golden Core Award from IEEE.

Nagender Aneja is a Collegiate Associate Professor at the Bradley Department of Electrical and Computer Engineering at Virginia Tech, Blacksburg, VA. He previously worked as a Research Scholar for the Department of Computer Science at Purdue University, West Lafayette, IN, and as Associate IP Lead for Microsoft Patent Research Services at CPA Global India. He has held several academic positions in India and Brunei Darussalam.