SECURITY AND PRIVACY CHALLENGES IN FEDERATED LEARNING FOR ANALYSIS OF ELECTRONIC HEALTH RECORDS

DEPARTMENT OF COMPUTING SECURITY
GOLISANO COLLEGE OF COMPUTING AND INFORMATION SCIENCES
ROCHESTER INSTITUTE OF TECHNOLOGY

February 28, 2025

Tanishq Borse
Department of Computing Security
Rochester Institute of Technology
tb7223@rit.edu

Suvam Barui
Department of Computing Security
Rochester Institute of Technology
sb9895@rit.edu

1 Abstract

The utilization of Machine Learning (ML) with Electronic Health Records (EHR) is becoming increasingly popular as a way to extract valuable information that can enhance decision-making in the healthcare industry. Such techniques necessitate the creation of high-quality learning models based on diverse and extensive datasets, which are challenging to obtain because of the sensitive nature of medical data from patients. Federated Learning (FL) is a methodology that enables the distributed training of machine learning models with datasets hosted remotely, without needing to collect data and therefore risking its privacy. FL is a promising approach to improving machine learning-based systems, bringing them more in line with regulatory requirements, enhancing trustworthiness, and maintaining data sovereignty. This survey paper presents a comprehensive literature review of current research on FL in the context of EHR data for healthcare applications. The study highlights key research topics, proposed solutions, case studies, and machine learning methods. Additionally, this article outlines a general architecture for FL applied to healthcare data based on the main insights gained from the literature review. The study reviews the extensive research on the privacy and security aspects of federated learning on electronic healthcare records.

2 Introduction

The digital transformation of the healthcare industry has brought about significant benefits, such as improved care coordination and increased efficiency. Electronic Health Records (EHRs) [1] are a vital part of health industry and contain a patientâs medical history, diagnoses, medications, treatment plans, immunization dates, allergies, radiology images, and laboratory test results, and allow access to evidence-based tools that providers can use to make decisions about a patientâs care. To facilitate the development of evidence-based tools that providers can use to make decisions about a patientâs care, sharing of electronic health data for collaborative analysis of larger and more diverse datasets is being practised. This can lead to significant advancements in medical research and the development of effective treatments for diseases such as cancer and other chronic illnesses.

2.1 Motivation

Machine Learning (ML) on EHRs can be used to collect datasets from various sources and store them in a central data warehouse, allowing machine learning models to be trained using the consolidated dataset. The utilization of centralized machine learning (ML) approaches is often hindered by numerous threats, such as privacy and security issues. If there are failures in the central disk, network, or links, it may not be possible to retrieve the central data. In the medical field, this can create significant issues as the data accessed by healthcare providers is highly critical. For instance, data vulnerability to cyberattacks is a significant threat, and vital data may be lost. Additionally, in centralized ML, data training is performed using data from a single server source, which can lead to sub-optimal disease prediction.

To overcome the issues of data privacy and sharing among healthcare industry, federated learning (FL) is a promising solution. With FL, model training is performed on the device

level, where each device is trained on its data and sends updates to the central servers. The central servers aggregate the updates and then send them back to the devices. FL offers higher prediction accuracy than centralized ML models and enables personalization in model learning. However, sharing data raises significant privacy concerns. FL privacy-preservation techniques are superior to those of centralized ML since it is not feasible to collect such extensive data in centralized ML and ensure its security[2].

This research has the potential to significantly improve the privacy and security of patient data, and inform the development of policies and guidelines to support the use of federated learning in healthcare. It can also aid healthcare providers and policymakers in making informed decisions about machine learning in healthcare, ultimately contributing to a more secure and efficient healthcare system that benefits both patients and providers.

2.2 Background

Protecting patient data privacy while enabling data sharing for research and clinical purposes is a significant challenge in the healthcare industry. The conventional approach to data sharing involves aggregating data from multiple sources into a centralized repository, which can raise privacy concerns due to the risk of data breaches and unauthorized access.

Federated learning provides an alternative solution by allowing institutions to collaborate on machine learning model development without sharing the underlying data. Instead, the data remains stored locally at each institution, and only model updates are shared between the participating entities. This approach has been successfully applied in other industries, such as finance and telecommunications, to improve data privacy and security while enabling collaborative analysis.

In the context of healthcare, the adoption of federated learning has been slow, partly due to the complexity of healthcare data and regulations governing data sharing. Nevertheless, recent studies have shown promising results for federated learning in healthcare applications, including disease prediction, clinical decision support, and medical imaging analysis. Distributed EHR, also known as decentralized EHR, is an emerging approach to EHRs that distributes patient data across multiple nodes or institutions, rather than storing it in a centralized repository. This approach can enhance patient privacy and data security by minimizing the risk of data breaches and unauthorized access. Additionally, it can enable efficient data sharing for research and clinical purposes, making it a potential use case for federated learning in healthcare.

Overall, exploring the feasibility and security challenges of federated learning in healthcare can identify the best practices for implementing this technique in the healthcare industry. This investigation can optimize the performance of federated learning while ensuring the confidentiality and integrity of patient data, ultimately contributing to the development of a more secure and efficient healthcare system.

2.3 Research Questions

- What are the security challenges associated with implementing federated learning in Electronic Health Records, and how to overcome them?
- What are Privacy issues in using federated learning in electronic healthcare records? What are the factors that need to be taken care of while deciding on the privacy budget and its efficiency?

- What are the ethical considerations and underlying risks of using federated learning for healthcare data governance and regulation?
- How to remediate the security and privacy challenges faced in Federated Learning to analyze Electronic Health Records?

  In conclusion, the goal of this study is to discover security and privacy challenges in federated learning for the analysis of Electronic Health Records(EHR) Data.

3 Literature Review

3.1 Federated Learning

Federated learning is a decentralized machine learning technique that enables multiple entities to collaboratively train a model without sharing the underlying data. Instead, the model is trained on local data on each participant's device, and only model updates are shared between the participating entities. This approach offers significant benefits over traditional centralized machine learning, which involves aggregating data from multiple sources into a central repository.

Federated learning is an emerging approach for analyzing healthcare data from multiple institutions. The traditional method of analyzing data from a single institution may be limited by institutional policies, available resources, and the characteristics of the local population. The benefits of analyzing data from multiple institutions include larger and more diverse datasets and an increased ability to develop and validate models that work across multiple institutions. Two main approaches to sharing data across institutions are aggregation and federation. In an aggregated network, data is physically brought together in a single location, but many institutions are reluctant to give up control of their data in this way. In a federated network, the data remains at the contributing institution, and queries are sent out to the data, rather than the data being brought to a central location. TriNetX, a commercial company, has built and operates the largest global research network of real-world clinical data using a federated approach. However, for a federated network to function effectively, the data needs to be harmonized both syntactically and semantically, and participating systems must be interoperable. The use of federated learning has significant implications for healthcare research, including improved efficiency, increased data privacy, and enhanced data quality. [3]

One of the primary advantages of federated learning is its ability to preserve data privacy and confidentiality. By keeping the data on local devices, federated learning avoids the risks associated with centralized data storage, such as data breaches and unauthorized access. This makes it an attractive option for industries that handle sensitive data, such as healthcare and finance. Another advantage of federated learning is its efficiency. Since the model is trained locally, there is no need to transfer large amounts of data to a central location, which can be time-consuming and resource-intensive. This also reduces the risk of data loss or corruption during transfer. Additionally, federated learning can help overcome the problem of data silos, where different institutions have access to different datasets, by allowing them to collaborate and learn from each other's data.

The concept of federated learning has been around for several years, but recent advances in machine learning algorithms and mobile computing have made it more practical and scalable. Google first popularized the idea of federated learning with its application

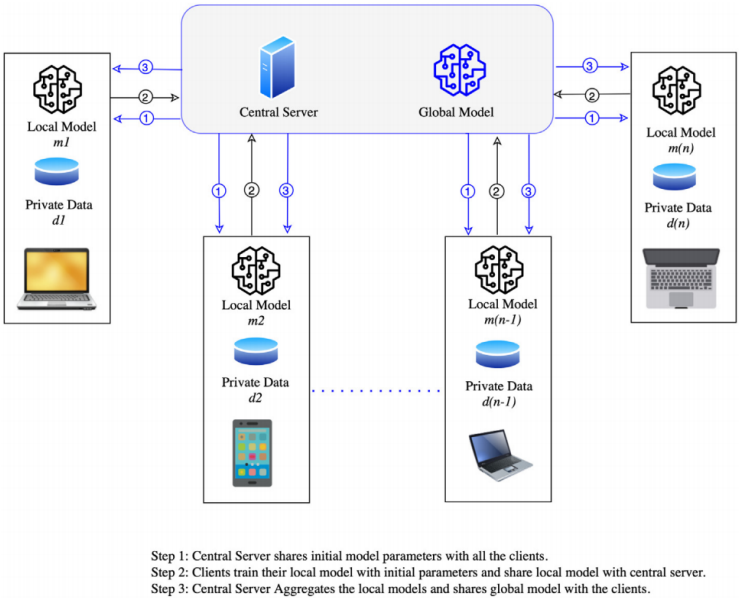

Figure 1: FL process flow

in mobile devices for improving keyboard predictions and text completion. Since then, federated learning has been applied in various other domains, including healthcare, finance, telecommunications, and energy management. In healthcare, federated learning has shown promising results for applications such as disease prediction, clinical decision support, and medical imaging analysis. By enabling multiple institutions to train models collaboratively, federated learning can improve the accuracy and generalization of models while preserving patient privacy. [3] [4]

Despite the advantages of federated learning, there are also some challenges associated with this approach. One of the main challenges is ensuring the quality and consistency of local data, which can vary significantly across participants. Another challenge is optimizing the federated learning algorithms to handle the heterogeneity and complexity of healthcare data, which can include structured and unstructured data from various sources. In conclusion, federated learning offers a promising approach to privacy-preserving machine learning in industries that handle sensitive data. While there are challenges associated with this approach, recent advances in algorithms and mobile computing have made it more practical and scalable. Further research is needed to optimize federated learning for healthcare applications and to address the challenges associated with data quality and consistency.

In an FL system, there are generally two main roles: (1) clients that hold their local datasets and (2) a server that orchestrates the entire training process and updates the global model without accessing the client datasets. The number of clients can be exceptionally large whereas there is usually only one server. In a special FL setting, the role of the server will be played by certain clients during the training phase. The FL training process generally consists of three key steps

- Step 1: FL initialization on the server side. First, the server initializes the weights of the global mode and the hyperparameters (e.g., the number of FL rounds, the total number of clients, and the number of clients to be selected during each training round). It then activates the clients, broadcasts the initialized global model, and

distributes calculation tasks to certain selected clients.

- Step 2: Local model training and update on the client side. First, the selected clients receive the current global information (e.g., weights or gradients) sent by the server and update their individual local model parameters using their local datasets, using the index of the current iteration round. Then, after finishing the local training, they send their local information (e.g., weights or gradients) to the server for model aggregation. During the local training phase, the goal of the selected client is to obtain the optimal local model parameters by minimizing the loss function.

- Step 3: Global model aggregation and update on the server side. The server first aggregates the received local information sent by the selected clients and then sends back the updated information to the clients for the next round of training. The goal is to obtain optimal global model parameters by minimizing the global loss function.

4 Methodolgy

4.1 Data

Medical datasets undergo various checks to protect sensitive information. For instance, the Health Insurance Portability and Accountability Act (HIPPA) has rules for creating limited datasets, which control identifiable information. Electronic Health Records (EHRs) are one type of medical record that stores a significant amount of medical and demographic data. EHRs provide a comprehensive view of a patient's care, including medical history, diagnoses, treatments, and laboratory results. However, EHRs contain sensitive information, so they are not widely available. Two common EHR datasets are the eICU Collaborative Database and the MIMIC Critical Care Database.

While various models have been developed using data from randomized controlled trials (RCTs) and population cohort studies, these datasets represent highly selective populations and may not be generalizable. In contrast, real-world data collected routinely as part of daily practice has several advantages. EHRs capture the entire patient journey, including changes in clinical state over time, as recorded by healthcare providers. This information is available to every person using the healthcare system and is more representative of the diverse patient population. Additionally, since this data is routinely collected at the point of care, it is readily available and exists in every healthcare system that uses EHRs. Therefore, building cancer prediction models leveraging EHR data is logical when envisioning their future use-case scenario. Developing the model in the same setting in which it will be clinically implemented, i.e., in the physicianâs office within the electronic patient medical record, deals upfront with some of the challenges inherent to this type of data source and helps overcome these challenges during the development and testing stages.

Federated learning (FL) is a machine learning (ML) approach that allows multiple parties to collaboratively train a model without sharing their raw data. Instead, each party trains the model on their local data and then shares only the model updates with a central server, which aggregates the updates to create a global model. This approach enables data privacy and security, while still allowing for the benefits of collaborative learning.

4.2 Existing Applications of Federated Learning on Electronic Health Records

The increasing adoption of EHRs in healthcare institutions has led to numerous studies exploring the application of machine learning to biomedical research [5]. However, using EHRs for machine learning poses challenges, including a lack of data and poor model generalizability. In particular, small hospitals may not have sufficient data for machine learning models to learn meaningful patterns for research projects involving rare diseases or conditions. Moreover, machine learning models trained on data from a single source may not generalize well and may perform poorly in different contexts. Federated learning (FL) is a promising approach to address these issues. FL allows training a global model on a larger and more diverse set of EHRs from multiple institutions while preserving privacy and improving the model's external validity. Studies have explored the effectiveness of FL for solving healthcare problems using EHRs, which can be categorized into predictive modeling and representation learning.

Numerous studies have achieved success in applying FL to predictive modeling on EHRs. For instance, Sharma et al. [6] proposed an FL framework for predicting in-hospital mortality for patients in the ICU, and Vaid et al.[7] employed FL to predict 7-day mortality for hospitalized COVID-19 patients using data from five different hospitals. Boughorbel et al. [8] presented an algorithm based on FedAvg, called federated uncertainty-aware learning algorithm (FUALA), to predict preterm birth in the context of distributed EHRs. Brisimi et al. [9] proposed an iterative cluster Primal-Dual Splitting (cPDS) algorithm for predicting hospitalizations due to cardiac events in FL settings. Huang et al. [10] proposed an FL algorithm called LoAdaBoost to predict the mortality of patients admitted to the ICU based on drugs prescribed during the first 48 h of their ICU stay. Pfohl et al. [11] conducted a comprehensive study to evaluate the efficacy of FL and differential privacy versus centralized training in predicting prolonged length of stay and in-hospital mortality across thirty-one hospitals. Grama et al. [12] evaluated the performance of different robust FL aggregation methods on two disease prediction tasks, diabetes mellitus onset prediction and heart failure prediction. Tan et al. [13] proposed a tree-based FL method for treatment effect estimation and used it to study the effect of oxygen saturation on hospital mortality among ICU patients with respiratory diseases. Tuladhar et al. [14] presented an ensemble approach to distributed learning of machine learning models for rare disease detection. Xue et al. [15] introduced a federated reinforcement learning system that employs Double Deep Q-Network (DDQN) to support personalized clinical decisions using data from smart devices at the edge as well as electronic medical records (EMRs) [16].

4.3 Techniques and Architectures in federated learning

There are several different techniques and architectures for implementing federated learning. Some of the most commonly used techniques are:

Federated Averaging: Federated Averaging (FedAvg) is a popular technique for Federated Learning (FL) that involves each participating device training a local model on its own data and then sending updated model parameters to a central server. The server aggregates the model updates using averaging and sends the updated global model back to the devices for further training until convergence is achieved. FedAvg enables distributed training with a large number of clients while maintaining data privacy by allowing clients to retain their data locally. However, a centralized approach requires extensive communication between the central server and clients, leading to possible channel blocking and privacy breaches in case of an attack. Decentralization can alleviate the communication burden on the

central server by enabling nodes to communicate only with their neighbors. There are multiple types of federated averaging techniques that can be used for preserving privacy in electronic health care records, such as FedSGD, FedProx: Proximal Federated Learning Framework, FedAvgM, and FedAdagrad. [17]

Federated Learning with Differential Privacy: Federated learning using differential privacy is an essential technique in privacy-preserving federated learning, particularly for electronic health records. The core concept behind differential privacy is to add random noise to data to conceal individual data points while preserving the statistical accuracy of the aggregated data. In federated learning using differential privacy, local models are trained on client data with added noise to protect privacy. The central server aggregates the models while adding additional noise to enhance privacy protection. This approach offers robust privacy guarantees while allowing collaborative model training. The use of federated learning with differential privacy has the potential to facilitate the sharing of electronic health records across multiple healthcare organizations, leading to improved medical research and decision-making while preserving privacy and security. As the healthcare industry increasingly adopts differential privacy, it enables the sharing of patient data while maintaining confidentiality, enabling more precise medical research and personalized treatment.

Secure Aggregation: The secure aggregation technique for federated learning in electronic healthcare records involves aggregating the model updates of clients in a privacy-preserving manner. This is achieved through the use of secure multi-party computation, where each client encrypts their updates before sending them to the server, which can perform computations on the encrypted data. The server then decrypts the result, ensuring that the privacy of each client's data is maintained. This approach enables the collaborative training of a global model without compromising the privacy and security of patient data.

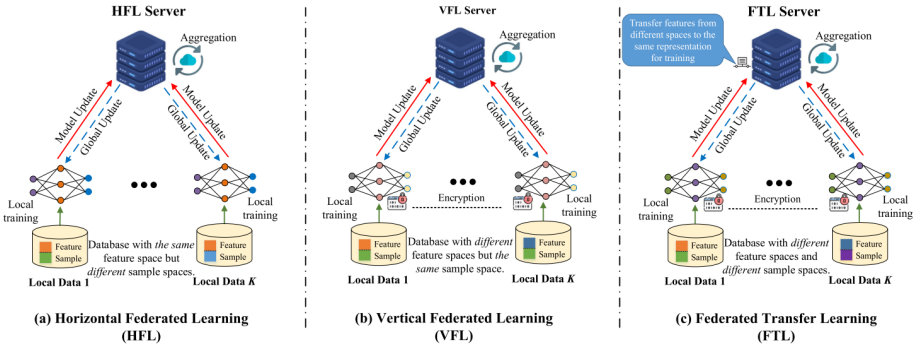

Figure 2: Common Categories of FL used in smart healthcare.

Cross-silo Federated Learning: This is an architecture used for collaborative machine learning when data is distributed across multiple servers. This approach involves partitioning data from different sources across various servers, where each server trains a local model on its own data and shares the model updates with a central aggregator. The cross-silo federated learning technique is often used in settings where there is no single central authority or organization, making it an ideal approach for scenarios such as inter-organizational collaboration or situations where data is sensitive and needs to be partitioned across different servers. By allowing the sharing of model updates between servers, cross-silo federated learning enables the collaborative training of machine learning

models while maintaining data privacy and security. [18]

Vertical Federated Learning: This architecture involves partitioning data horizontally across multiple devices or servers, where each device trains a local model on its own data and shares model updates with a central server. It is commonly used in situations where a central authority or organization manages the data. VHL focuses on federated training of health datasets that have the same sample space with different data feature spaces, as shown in Figure 2(b). To address the issue of data sample overlapping at distributed clients, entity alignment solutions can be used in conjunction with encryption techniques during local training. An example of VFL in IMoT applications is the shared learning model among entities in a smart healthcare environment, such as hospitals and insurance companies. In this context, hospitals and insurance companies (which have different data features) that serve patients (who are in the same sample space) can participate in a VFL process to collaboratively train an AI model using their datasets, such as historical medical records at hospitals and healthcare costs at the insurance company, to make intelligent healthcare decisions. [19]

Federated Transfer Learning: In this architecture, systems are designed to handle datasets with varying sample spaces and feature spaces. Figure 2(c) illustrates this concept. FTL utilizes transfer learning techniques to map feature values from different feature spaces to a common representation, which is then used to train local datasets. Additional privacy protection can be achieved by using encryption techniques such as random masks during gradient exchange between clients and the server. In the context of smart healthcare, FTL can facilitate disease diagnosis by enabling collaboration between countries with multiple hospitals that serve patients with distinct therapeutic programs. By enriching the shared AI model output, FTL can help improve the accuracy of diagnoses.[20]

Horizontal Data Partitioning: The healthcare clients can participate in training a shared global model using their datasets, which own the same feature space while having different sample spaces, as shown in Figure 2(a). In this regard, local FL participants can adopt the same AI model (e.g., neural network-NN) for training their datasets. Subsequently, the server will combine the local updates transmitted from local participants to build a global update without the need for direct access to local data. An HFL example in smart healthcare can be the detection of speech disorders where multiple users speak the same sentence (feature space) with different voices (sample space) on their smartphones and then the local speaking updates are averaged by a parameter server to create a global model for speech recognition.

4.4 Security vulnerabilities in Federated Learning

Federated learning has gained a lot of attention in recent years due to its potential to improve data privacy and security while enabling collaborative analysis. However, since the nature of healthcare records is highly sensitive, the security risks associated with federated learning are concerning. In this survey paper, we aim to provide an overview of the existing research on attacks and defenses in federated learning systems.

To establish a more secure environment, it is essential to identify vulnerabilities and take preventive measures to protect against potential risks. Neglecting the protection of personally identifiable information (PII) or violating data protection regulations could not only harm the reputation of an organization but also result in legal consequences. It is compulsory for FL developers to conduct vulnerability assessments and implement robust

defenses to safeguard data privacy and security. To obtain a deeper understanding of vulnerabilities, we have classified the sources of vulnerabilities in the FL process. Our findings indicate that there are four distinct sources that may be exploited as weak points. These sources are listed below.

- Communication Protocol: The process of FL involves a significant amount of communication over a network, as it follows an iterative learning approach with randomly selected clients. To ensure the anonymity of the source and message content during communication, a mixed network based on public-key cryptography is suggested. However, as FL involves more rounds of training, a non-secure communication channel can become a vulnerability that may be exploited.
- Client Data Manipulation: In a larger landscape, FL involves numerous clients that are susceptible to data reconstruction and model parameter attacks by attackers. Moreover, access to the global model can also be vulnerable to exploitation.
- Compromised Central Server: The central server in FL is responsible for sharing initial model parameters, aggregating local models, and sharing global model updates to all clients. Therefore, the server should be robust and secure to prevent curious attackers from exploiting its open vulnerabilities. Whether the server is cloud-based or physical, it must be checked to ensure that it is secure.
- Weaker Aggregation Algorithm: The aggregation algorithm is the central authority in FL, as it handles the update of the local model. Thus, it must be intelligent enough to detect abnormality with client updates and have a configuration to drop updates from suspicious clients. Failing to configure a standardized aggregation algorithm can make the global model vulnerable to attacks.

4.5 Security threats/attacks in Federated Learning

FL can be vulnerable to attacks where a malicious agent exploits vulnerabilities to manipulate the global model. In such cases, the attacker targets various clients to gain access to local data, training procedures, hyper-parameters, or updated weights in transit. These can be modified to launch attacks on the global model. Different security threats and attacks are classified and described in the following sections.

4.5.1 Poisoning Attacks

The concept of poisoning attacks is a major security threat in FL, as each client in FL has access to the training data, making it highly susceptible to tampered data weights added to the global ML model. Poisoning attacks can occur during the training phase and can impact either the training dataset or the local model, indirectly affecting the global ML model performance. Poisoning attacks can target various artifacts in the FL process, including data and model poisoning, and data modification [21].

- Data Poisoning: FL is highly susceptible to data poisoning attacks, as malicious clients can actively contribute to training data and manipulate the training process. Data poisoning is defined as generating dirty samples to train the global model in hopes of producing falsified model parameters and sending them to the server. Data injection, a subcategory of data poisoning, involves the malicious client injecting malicious data into a client's local model processing, ultimately manipulating the global model with malicious data.

- Model Poisoning: In model poisoning, the malicious party modifies the updated model before sending it to the central server for aggregation, easily poisoning the global model. Recent research suggests that model poisoning attacks are more effective than data poisoning attacks, especially in a large-scale FL product with many clients.
- Data Modification: Data modification attacks involve changing/altering the training dataset to confuse the ML model. Techniques include feature collision, random label swap, and data injection, and they can be considered a type of ML data poisoning attack in FL.

4.5.2 Inference

Inference attacks pose a significant risk to privacy. Although this type of attack is not the main focus, it is still included here to provide a comprehensive overview of the threats in FL. Inference attacks are as serious as poisoning attacks, given the high likelihood of these attacks being carried out either by the participants or a malicious centralized server involved in the FL process[22].

4.5.3 Backdoor Attacks

Compared to poisoning and inference attacks, backdoor attacks are less transparent. Backdoor attacks involve injecting a malicious task into an existing model without compromising the accuracy of the actual task. Detecting backdoor attacks is challenging because the accuracy of the ML task may not be immediately affected. To mitigate the risks of backdoor attacks, using model pruning and fine-tuning is suggested. Backdoor attacks are severe because it takes a significant amount of time to detect their occurrence, and their impact is high. Backdoor attacks have the potential to confuse ML models and confidently predict false positives. Trojans threats are a similar type of backdoor attack that aim to perform a malicious task in stealth mode while retaining the existing task of the ML model.

4.5.4 GANs

The use of Generative Adversarial Network (GAN)-based attacks in FL has been studied by several researchers [23], and their findings indicate that such attacks pose a significant threat to the security and privacy of the system. The research presented in [24] shows how GANs can be used to obtain training data through inference and how GANs can be used to poison the training data. Since it is difficult to anticipate all the potential risks of GAN-based threats, such attacks are considered a high-impact and prioritized threat.

4.5.5 Malicious server

The central server plays a crucial role in cross-device FL, as it performs a majority of the tasks, such as selecting the model parameters and deploying the global model. However, a compromised or malicious server can have a significant impact on the system's security. Even an honest server with curiosity or malicious intent can extract private client data or manipulate the global model, utilizing the shared computational power to build malicious tasks into the global ML model.

Table 1
Threats in FL.

Threats	Severity	ML Framework	Source of Vulnerability
Poisoning	High	DML/FL	Client Data Manipulations, Compromised Central Server
Inference	High	FL	Client Data Manipulations, Compromised Central Server
Backdoor Attacks	High	DML/FL	Client Data Manipulations
GANs	High	FL	Client Data Manipulations, Compromised Central Server
Malicious Server	High	DML/FL	Compromised Central Server
Communication bottlenecks	High	DML/FL	Weaker Communication bandwidth
Free-riding	Medium	FL	Clients in FL
Unavailability	Medium	FL	Clients in FL
Eavesdropping	Medium	FL	Weaker Communication Protocol
Interplay with data protection laws	Low	FL	Implementer's of FL Environment
System disruption IT downtime	Low	FL	Clients and Centralized Server in FL

Figure 3: Attacks and Threats in FL

4.5.6 Communication bottlenecks

Communication bandwidth is a significant challenge when training an ML model from data collected on multiple heterogeneous devices. In order to reduce communication costs, the FL approach involves transferring trained models instead of sending large amounts of data. However, it is still important to preserve communication bandwidth, and there are several algorithms that are based on asynchronous aggregation of models, as well as strategies that can perform well even with low-communication bandwidth. Several research studies [25], [26] have been conducted on preserving communication bandwidth in FL environments. This threat is considered to be high in severity, as communication bottlenecks can significantly disrupt the FL environment.

4.5.7 Eavesdropping attacks

In an eavesdropping attack, The adversaries located in the communication channel between the central server and local workers can launch eavesdropping attacks. The adversaries can steal or tamper with some meaningful information, such as model weights or gradients, in each communication. Eavesdropping attacks can lead to the leakage of sensitive information and the compromise of the privacy of the users. [21]

4.5.8 Sybil Attacks

In a Sybil attack, an attacker creates multiple fake identities and uses them to participate in the federated learning process, aiming to bias the model's training towards the attacker's objectives. Sybil attacks can cause the model to produce incorrect outputs when processing new data.[27]

4.5.9 Reconstruction Attacks

In a reconstruction attack, an attacker uses the model's output to reconstruct the training data set. Reconstruction attacks can lead to the leakage of sensitive information and the compromise of the privacy of the users[18] Deep Leakage from Gradient (DLG) [28] was the first exploration to fully reveal the private training data from gradients, which can obtain the training inputs as well as the labels in only a few iterations.

4.6 Defensive techniques for Security Vulnerabilities in Federated Learning

Defense techniques play a crucial role in protecting against known attacks and reducing the probability of risks. These techniques can be classified as proactive or reactive. Proactive defense involves predicting potential threats and employing cost-effective defense strategies, whereas reactive defense involves deploying a defense technique as a patch-up in the production environment when an attack is identified. In addition to existing defense techniques, researchers are exploring new add-on technologies and algorithms to enhance FL security capabilities. One such example is the integration of FL with blockchain technology, as demonstrated in many research works. This technology serves two main purposes in FL: providing incentives to major contributors to the global ML model and ensuring the security of the global model by saving its parameters and weights on a blockchain ledger. Some studies have shown how blockchain technology can promote coordination and trust among federated clients, while ongoing research is investigating incentive mechanisms to encourage clients to contribute proactively to the learning process.

4.6.1 Sniper

The proposed Sniper approach in [29] aims to mitigate the threat of distributed poisoning attacks in FL by detecting legitimate users and decreasing the success rate of poisoning attacks, even when multiple attackers are involved. This is important because although poisoning attacks in a centralized setting have been well studied, the effectiveness of distributed poisoning with multiple attackers is still unclear. Sniper outperforms existing defense mechanisms against distributed poisoning attacks and is effective in reducing the impact of poisoning attacks. Therefore, although distributed poisoning attacks may pose a significant threat in FL, there are promising defense techniques like Sniper that can help mitigate these attacks.

4.6.2 Knowledge Distillation

Knowledge distillation is a process where a small model is trained to mimic the behavior of a larger, more complex model by transferring its knowledge. This technique can be used to save computational cost involved in training a model. In the context of FL, knowledge distillation can be used to enhance the security of client data by sharing knowledge instead of model parameters. The authors of [30] proposed a federated model distillation framework that allows for the use of personalized ML models and uses translators to collect knowledge to be shared with each client. This approach can help to reduce the amount of sensitive data that needs to be transmitted between clients and the central server, thereby improving the privacy and security of the FL system.

4.6.3 Anomaly Detection

Anomaly detection techniques are an important defense mechanism in FL that can be used to detect various types of attacks such as data and model poisoning attacks or trojan threats. These techniques utilize statistical and analytical methods to identify events that do not conform to an expected pattern or activity. To effectively detect attacks, an anomaly detection system requires a profile of the normal behavior or events to detect deviations from the normal behavior profile.

Table 2
FL defense Techniques.

Defenses	Description	Threats
Sniper	configure euclidean distance check on global server to exclude adversarial updates	Poisoning
Knowledge distillation	Transferring knowledge from fully trained model to another model	Eavesdropping Inference GANs
Anomaly Detection	Monitoring for suspicious updates of clients	Poisoning Trojans Model update poisoning
Moving target defense	Obfuscating source of vulnerability	Eavesdropping
Pruning	Reduce the size of Neural network model	Backdoor Attacks Model Computation Communication costs
Data Sanitization	removing/deleting the data after use	Poisoning Attacks
Trusted execution Environment	Provides integrity and confidentiality of the code executed on a server	Malicious server
Fools Gold	Based on the diversity of client updates sybil attacks are identified	sybil-based label flipping backdoor poisoning attacks
Federated Multi-task Learning	Train models for multiple related tasks simultaneously	Device drop Fault tolerance

Figure 4: Defense Techniques against attacks/threats in FL

4.6.4 Moving Target Defense

In FL, the concept of moving target defense can also be applied to enhance security. The authors in [31] propose a dynamic FL architecture where the server and clients change their roles frequently during training to prevent the attackers from identifying and exploiting vulnerable clients. In [32], the authors propose a mechanism called Differential Privacy-Preserving Moving Target Defense (DPMTD), which enhances the privacy protection of client data by constantly changing the weights of the global model during training. A multi-agent reinforcement learning approach is proposed for FL where the agents' roles are constantly changing, making it difficult for attackers to predict the behavior of each agent. This approach is effective in protecting against model poisoning attacks. In summary, moving target defense is an effective defense technique that can be used in FL to enhance security by constantly changing the system's configuration, roles, or weights, making it harder for attackers to exploit vulnerabilities.

4.6.5 Pruning

Pruning is a technique that removes the unnecessary connections, weights, and neurons from a pre-trained neural network without affecting its accuracy. This results in a smaller, more compact model that requires less computational power and memory to perform inference, making it ideal for deployment in resource-constrained environments like FL. Pruning can be done at different levels of granularity, such as pruning individual weights, entire neurons, or entire layers. The authors in [33] propose a federated model pruning technique where each client prunes its local model based on a pre-defined threshold and only sends the pruned model updates to the server for aggregation. Similarly, the authors in [34] propose a decentralized pruning technique where clients communicate with each other to exchange pruning information and collectively decide which neurons to prune. This approach not only reduces communication overhead but also provides additional security against attacks as clients do not have access to the entire model.

4.6.6 Data Sanitization

Data sanitization is a technique used to detect and filter out anomalous or suspicious data points from the training data. It is a common defense technique used against data

poisoning attacks in FL environments. The idea is to identify and remove malicious data that might be introduced by adversarial clients. The technique was first proposed by [35], and recent work has aimed to improve it by utilizing robust statistics models.

4.6.7 Foolsgold

In the federated learning (FL) environment, malicious clients can compromise the security by creating multiple fake identities and sending false updates to the central server. To address this issue, the authors of [36] suggest a Foolsgold approach that can effectively defend against various types of attacks, such as Sybil-based attacks, label flipping attacks, and backdoor poisoning attacks.

4.6.8 Federated MultiTask Learning

Federated Learning is a collaborative machine learning method that enables the training of models on decentralized mobile devices while preserving their local data privacy. This approach can be extended to federated multi-task learning, which enables personalized but shared models among devices. The MOCHA Framework, proposed in [37], is designed to accelerate the learning process while addressing statistical and system challenges such as high communication costs, stragglers, and fault tolerance issues in the FL environment. The framework is also capable of handling system heterogeneity and being resilient to system drop, as shown by their experiments.

4.6.9 Trusted Execution Environment

A Trusted Execution Environment (TEE) is a secure environment for code execution, often used for privacy-preserving purposes in various ML models. It is especially useful in the context of federated learning, where computing resources are limited. The TEE provides a secure area of the main processor for executing code, ensuring tamper-resistance, integrity, and confidentiality of the executed code.

4.7 Privacy in Federated Learning

Federated Learning (FL) aims to preserve user privacy by reducing the amount of user data that is transmitted to a central server. Despite the privacy benefits of FL, it is not completely secure from attacks, and its enabling technology is still developing. Therefore, this section explores the privacy issues and the current technological advancements in FL, with the intention of providing more information for future development.

4.8 Privacy threats/attacks in FL

Federated Learning (FL) intends to ensure the privacy of participants by requiring them to share local training model parameters instead of their actual data. Nevertheless, studies indicate that FL is still vulnerable to privacy threats as adversaries can use the uploaded parameter to partially reveal each participant's training data in the original training dataset. These privacy risks in FL fall under various categories of inference-based attacks.

4.8.1 Membership inference attacks

Inference attacks are a type of attack that attempts to uncover training data details. One specific type of inference attack is the Membership Inference attack which checks whether or not data exists in a training set. Attackers can misuse the global model to gain information on the training data of other users. This is accomplished by training a predictive model that guesses the original training data through inference. Researchers have explored the vulnerability of neural networks to memorize their training data, which makes them susceptible to both passive and active inference attacks.

4.8.2 Unintentional data leakage reconstruction through inference

The unintended information leakage in Federated Learning happens when clients' updates or gradients reveal confidential information at the central server. In [38], researchers successfully conducted an inference attack to reconstruct data of other clients by exploiting this vulnerability.

Another study [39] examines how GANs-based inference attacks can reveal private data of an honest client. In this case, the adversary generates data that is similar to the training data and retrieves sensitive information from other clients. In [40], malicious or curious clients use global model parameters to reconstruct the training data of other clients.

4.8.3 GANs-based inference attacks

Generative adversarial networks (GANs) have gained popularity in recent years, and they can also be applied to federated learning (FL) approaches. In [23], the mGAN-AI framework is proposed for GAN-based attacks on FL. This framework's passive version analyzes all client inputs, while the active version isolates a client by sending the global update only to the isolated instance. The inference attack achieves the highest accuracy with the mGAN-AI framework because it does not interfere with the training process. However, FL clients may act as potential adversaries, using old local data as their contribution in exchange for the global model, which they can then use to deduce other clients' information using inference techniques. Such behaviors are challenging to distinguish due to the limited knowledge of clients' profiles and reputation. Additionally, collaborative training with parameter-only updates makes it difficult for the FL server to evaluate each client's contribution's effects.

4.9 Mitigation of threats and enhancing the general privacy-preserving feature of FL

The following section discusses the strategies that can be used to address the privacy risks identified in federated learning. One of the primary methods used to protect privacy in FL is the retention of data at the client level. To further enhance privacy and mitigate threats in FL, current algorithms primarily fall into two categories: Secure Multi-party Computation (SMC) and Differential Privacy (DP).

4.9.1 Secure multi-party computation

The Secure Multi-party Computation (SMC) concept is introduced to secure inputs in multi-participant computations. SMC is utilized in FL to secure updates from clients,

and it efficiently prevents data leakage of clients at the central server. SMC's challenge is the trade-off between efficiency and privacy. To mitigate the risk of client data exposure, the work in [41] combines homomorphic encryption and differential privacy. Client-level differential privacy and encrypting the model update ensure protection against the honest but curious server and other users of FL. SMC-based solutions need more time and may negatively affect model training and data freshness aware training tasks. Lightweight SMC solutions for FL clients are still an open problem.

4.9.2 Differential privacy

Differential Privacy (DP) is a technique used to add noise to personal sensitive attributes to preserve privacy. DP is used in Federated Learning (FL) to protect each user's privacy by adding noise to the parameters they upload. DP has been combined with Secure Multi-party Computation (SMC) to achieve high accuracy and secure FL models. DP is also implemented in other machine learning domains such as reinforcement learning and distributed ML. However, DP techniques can bring uncertainty to uploaded parameters and harm training performance, making it difficult for the FL server to evaluate client behavior.

4.9.3 VerifyNet

VerifyNet is a framework for federated learning that ensures privacy and security through a double-masking protocol, which makes it difficult for attackers to infer training data. It also allows clients to verify central server results to ensure reliability. In addition to these benefits, VerifyNet can handle multiple dropouts effectively. However, the framework has a communication overhead issue since the central server has to send verifiable proofs to each client.

4.9.4 Adversarial training

Evasion attacks can inject adversarial samples into FL models, affecting their robustness. Adversarial training is a defense technique that aims to make the FL model robust to known attacks. However, adversarial training is still vulnerable to black-box attacks, leading to the introduction of Ensemble Adversarial Training, which augments training data with perturbations. Other defense techniques include FEDXGB, which addresses user drop-out and leakage of private training data, and Anti-GAN and FedGP, which generate fake data at each client node to prevent inference attacks.

4.10 Evaluation of associated cost with the privacy-preserving techniques

Each improvement to privacy protection comes with its own costs and consequences. For example, Secure Multi-party Computation and Differential Privacy can enhance privacy protection in FL, but at the expense of reduced accuracy and efficiency. In Secure Multi-party Computation, cryptography-based methods require each client to encrypt all uploaded parameters, which can be computationally expensive and therefore a concern for IoT devices. Thus, enhancing privacy through encryption can compromise the efficiency of the ML model.

The authors of a study [42] aimed to evaluate the costs of federated learning (FL) and simulated the experiment using Reddit datasets to analyze the accuracy of FL models with differential privacy (DP) enabled. The results showed that the accuracy of the DP-FL and non-DP-FL models were similar for datasets with similar vocabulary size. However, for datasets with varying vocabulary size, the accuracy of DP-FL models was lower than that of non-DP-FL models. The study concluded that DP negatively affects the accuracy of FL models in heterogeneous environments.

Table 3
Approaches to enhance privacy preservation in FL.

Approach	Cost	Methodology
Secure Multi-party Computation	Efficiency loss due to encryption	Encrypt uploaded parameters
Differential Privacy	Accuracy loss due to added noise in client's model	Add random noise to uploaded parameters
Hybrid	Subdued cost on both efficiency and accuracy	Encrypt the manipulated parameter
VerifyNet	Communication overhead	Double-masking protocol Verifiable aggregation results
Adversarial Training	Computation power, training time for adversarial samples	Include adversarial samples in training data

Figure 5: Approaches to enhance privacy preservation in FL.

DP-based techniques introduce random noise to the parameters in order to protect privacy during communication and on the server. However, this noise inevitably affects the accuracy of the model and may also impact the convergence of the global aggregation. Therefore, there is a trade-off between the strength of privacy protection and the loss of accuracy and efficiency (convergence time). If the FL model prioritizes privacy, it will sacrifice accuracy and take more time to converge. Conversely, if the model needs to maintain a certain level of accuracy or convergence time, it must assess whether the privacy protection level is acceptable or not. Table 3 summarizes the privacy-preserving techniques discussed in this section along with their characteristics.

5 Conclusion

Federated learning is a promising solution for training machine learning models while preserving the confidentiality of sensitive data, particularly in healthcare applications where sharing Electronic Health Record (EHR) data is restricted. Although successful case studies have been reported, there are still several research questions to be addressed before FL can be widely used in healthcare. The purpose of FL is to expand the advantages of machine learning to domains that handle sensitive data. This study provides a thorough examination of the achievements, issues, and impacts regarding security and privacy in the FL environment. The aim of our evaluation and results is to provide fresh perspectives and draw the attention of the community to the development of secure and private FL environments that can be widely adopted. In the future directions section, we identify the areas within FL that require further investigation and research. As a relatively new framework in the market, FL needs further exploration to determine which enhancement techniques are most suitable for different FL environments.

References

[1] R. Kohli and S. S.-L. Tan, "Electronic health records: How can is researchers contribute to transforming healthcare?" *MIS Quarterly*, vol. 40, no. 3, pp. 553–574,

2016. [Online]. Available: https://www.jstor.org/stable/26629027

[2] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140 699–140 725, 2020.

[3] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang, "Federated learning for healthcare informatics," *Journal of Healthcare Informatics Research*, vol. 5, no. 1, p. 1â19, Mar 2021.

[4] S. M. Halim, L. Khan, K. W. Hamlen, B. Thuraisingham, and M. D. Hossain, "A federated approach for learning from electronic health records," in *2022 IEEE 8th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, May 2022, p. 218â223.

[5] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, vol. 19, no. 6, pp. 1236–1246, 05 2017. [Online]. Available: https://doi.org/10.1093/bib/bbx044

[6] P. Sharma, F. E. Shamout, and D. A. Clifton, "Preserving patient privacy while training a predictive model of in-hospital mortality," *arXiv preprint arXiv:1912.00354*, 2019.

[7] A. Vaid, S. K. Jaladanki, J. Xu, S. Teng, A. Kumar, S. Lee, S. Somani, I. Paranjpe, J. K. De Freitas, T. Wanyan *et al.*, "Federated learning of electronic health records to improve mortality prediction in hospitalized patients with covid-19: machine learning approach," *JMIR medical informatics*, vol. 9, no. 1, p. e24207, 2021.

[8] S. Boughorbel, F. Jarray, N. Venugopal, S. Moosa, H. Elhadi, and M. Makhlouf, "Federated uncertainty-aware learning for distributed hospital ehr data," *arXiv preprint arXiv:1910.12191*, 2019.

[9] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi, "Federated learning of predictive models from federated electronic health records," *International journal of medical informatics*, vol. 112, pp. 59–67, 2018.

[10] L. Huang, Y. Yin, Z. Fu, S. Zhang, H. Deng, and D. Liu, "Loadaboost: Loss-based adaboost federated machine learning with reduced computational complexity on iid and non-iid intensive care data," *Plos one*, vol. 15, no. 4, p. e0230706, 2020.

[11] S. R. Pfohl, A. M. Dai, and K. Heller, "Federated and differentially private learning for electronic health records," *arXiv preprint arXiv:1911.05861*, 2019.

[12] M. Grama, M. Musat, L. Muñoz-González, J. Passerat-Palmbach, D. Rueckert, and A. Alansary, "Robust aggregation for adaptive privacy preserving federated learning in healthcare," *arXiv preprint arXiv:2009.08294*, 2020.

[13] X. Tan, C.-C. H. Chang, L. Zhou, and L. Tang, "A tree-based model averaging approach for personalized treatment effect estimation from heterogeneous data sources," in *International Conference on Machine Learning*. PMLR, 2022, pp. 21 013–21 036.

[14] A. Tuladhar, S. Gill, Z. Ismail, N. D. Forkert, A. D. N. Initiative *et al.*, "Building machine learning models without sharing patient data: a simulation-based analysis of distributed learning by ensembling," *Journal of biomedical informatics*, vol. 106, p. 103424, 2020.

[15] Z. Xue, P. Zhou, Z. Xu, X. Wang, Y. Xie, X. Ding, and S. Wen, "A resource-constrained and privacy-preserving edge-computing-enabled clinical decision system: A federated reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9122–9138, 2021.

[16] T. K. Dang, X. Lan, J. Weng, and M. Feng, "Federated learning for electronic health records," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 5, jun 2022. [Online]. Available: https://doi.org/10.1145/3514500

[17] T. Sun, D. Li, and B. Wang, "Decentralized federated averaging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4289–4301, 2023.

[18] C. Wang, X. Wu, G. Liu, T. Deng, K. Peng, and S. Wan, "Safeguarding cross-silo federated learning with local differential privacy," *Digital Communications and Networks*, vol. 8, no. 4, p. 446â454, 2022.

[19] D. C. Nguyen, Q.-V. Pham, P. N. Pathirana, M. Ding, A. Seneviratne, Z. Lin, O. Dobre, and W.-J. Hwang, "Federated learning for smart healthcare: A survey," *ACM Computing Surveys*, vol. 55, no. 3, pp. 60:1–60:37, Feb 2022.

[20] S. Saha and T. Ahmad, "Federated transfer learning: concept and applications," *CoRR*, vol. abs/2010.15561, 2020. [Online]. Available: https://arxiv.org/abs/2010.15561

[21] P. Liu, X. Xu, and W. Wang, "Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives," *Cybersecurity*, vol. 5, no. 1, p. 4, Feb 2022.

[22] 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)*, 2017, pp. 3–18.

[23] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: User-level privacy leakage from federated learning," in *IEEE INFOCOM 2019-IEEE conference on computer communications*. IEEE, 2019, pp. 2512–2520.

[24] J. Zhang, J. Chen, D. Wu, B. Chen, and S. Yu, "Poisoning attack in federated learning using generative adversarial nets," in *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. IEEE, 2019, pp. 374–380.

[25] L. WANG, W. WANG, and B. LI, "Cmfl: Mitigating communication overhead for federated learning," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 2019, pp. 954–964.

[26] X. Yao, C. Huang, and L. Sun, "Two-stream federated learning: Reduce the communication costs," in *2018 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 2018, pp. 1–4.

[27] C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," 2020.

[28] L. Zhu and S. Han, *Deep Leakage from Gradients*. Cham: Springer International Publishing, 2020, pp. 17–31. [Online]. Available: https://doi.org/10.1007/978-3-030-63076-8_2

[29] D. Cao, S. Chang, Z. Lin, G. Liu, and D. Sun, "Understanding distributed poisoning attack in federated learning," in *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 2019, pp. 233–239.

[30] D. Li and J. Wang, "Fedmd: Heterogenous federated learning via model distillation," *arXiv preprint arXiv:1910.03581*, 2019.

[31] R. Colbaugh and K. Glass, "Moving target defense for adaptive adversaries," in *2013 IEEE International Conference on Intelligence and Security Informatics*. IEEE, 2013, pp. 50–55.

[32] S. Jajodia, A. K. Ghosh, V. Swarup, C. Wang, and X. S. Wang, *Moving target defense: creating asymmetric uncertainty for cyber threats*. Springer Science & Business Media, 2011, vol. 54.

[33] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *Research in Attacks, Intrusions, and Defenses: 21st International Symposium, RAID 2018, Heraklion, Crete, Greece, September 10-12, 2018, Proceedings 21*. Springer, 2018, pp. 273–294.

[34] Y. Jiang, S. Wang, V. Valls, B. J. Ko, W.-H. Lee, K. K. Leung, and L. Tassiulas, "Model pruning enables efficient federated learning on edge devices," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[35] G. F. Cretu, A. Stavrou, M. E. Locasto, S. J. Stolfo, and A. D. Keromytis, "Casting out demons: Sanitizing training data for anomaly sensors," in *2008 IEEE Symposium on Security and Privacy (sp 2008)*. IEEE, 2008, pp. 81–95.

[36] C. Fung, C. J. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," *arXiv preprint arXiv:1808.04866*, 2018.

[37] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," *Advances in neural information processing systems*, vol. 30, 2017.

[38] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE symposium on security and privacy (SP)*. IEEE, 2019, pp. 691–706.

[39] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the gan: information leakage from collaborative deep learning," in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, 2017, pp. 603–618.

[40] A. Bhowmick, J. Duchi, J. Freudiger, G. Kapoor, and R. Rogers, "Protection against reconstruction and its applications in private federated learning," *arXiv preprint arXiv:1812.00984*, 2018.

[41] M. Hao, H. Li, G. Xu, S. Liu, and H. Yang, "Towards efficient and privacy-preserving federated deep learning," in *ICC 2019-2019 IEEE international conference on communications (ICC)*. IEEE, 2019, pp. 1–6.

[42] E. Bagdasaryan, O. Poursaeed, and V. Shmatikov, "Differential privacy has disparate impact on model accuracy," *Advances in neural information processing systems*, vol. 32, 2019.