A Survey of Security Issues in Federated Laearning

1st Yunhao Feng
National University of Defense and Technology
Changsha, China
fengyunhaonudt@nudt.edu.cn

2nd Yinjian Hou
National University of Defense and Technology
Changsha, China
houyinjian18@nudt.edu.cn

Abstract—As people's awareness of the importance of personal privacy protection has grown, there has been a surge of interest in federated learning, which is a machine learning paradigm that enables training without requiring access to users' private data.

I. Introduction

The rapid development of digital technology has made the diversification, informationization, and diversity of digital data the main topics of the current era. Meanwhile, deep learning (DL) has demonstrated tremendous success in multiple fields, including computer vision, natural language processing, and graphic networks. Clearly, using diverse data in deep learning models can effectively improve their ability. However, there is also a growing interest in data privacy protection, such as the General Data Protection Regulation (GDPR) [1]. On the other hand, data sources may encounter the challenge of distributed storage, as is the case with data from mobile smart devices or Internet of Things (IoT) scenarios [2], [3]. Therefore, utilizing these data to train models requires overcoming limitations related to distribution and privacy [4].

To solve these problems, federated learning(FL) is a machine learning paradigm proposed as a possible response to these challenges [5]. FL enables collaborative model building among distributed members while ensuring sensitive data remains within each participant's control [6]. Specifically, federated learning allows two or more participants to collaboratively train a shared global DL model while keeping their training datasets locally. Each participant trains the shared model on its own training data and exchanges and updates model parameters with other participants. Federated learning can improve the training speed and the performance of the shared model while protecting privacy of the participants' training datasets [7]. Thus, it is a promising technique for the scenarios where the training data is sensitive (e.g., medical records, personally identifiable information, etc.) [8], [9].

Federated learning can be classified based on whether the participating datasets are the same, resulting in two types: homogeneous federated learning and heterogeneous federated learning [10], [11]. In homogeneous federated learning, all participants have datasets with the same characteristics and data distribution, whereas in heterogeneous federated learning, participants' datasets may differ in their characteristics and data distribution. The second

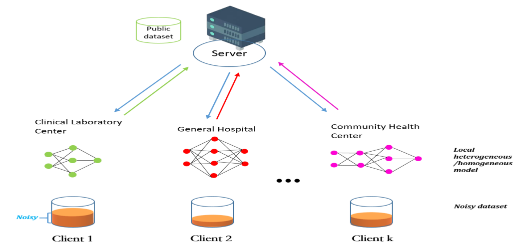

Fig. 1. A schematic of federated learning.

classification of federated learning is based on whether the models involved are the same, resulting in two types: horizontal federated learning and vertical federated learning [12], [13]. In horizontal federated learning, all participants have the same model architecture, but may have different local data [14], while in vertical federated learning, each participant has a different model architecture but they collaborate on processing the same set of data together [15]. The third way to classify federated learning is based on the type of task involved, resulting in several types such as federated learning for clustering [16], [17], federated learning for classification [18], [19], federated learning for regression [20], among others. The fourth way to classify federated learning is based on the optimization approach used between the participants, resulting in several types such as federated averaging [21], [22], federated learning optimization, federated meta-learning [23], and so on.

Federated learning methods currently face significant challenges related to their robustness. This article focuses on three main attacks. including backdoor attacks [24], [25], [26], [27], [28], adversarial attacks [31], [32], [33], [34], and Byzantine attacks [29], [30]. A backdoor attack involves a malicious participant in the federated learning process adding a backdoor to the model being trained, which can be triggered by a specific input pattern, allowing the attacker to control the output of the model in a targeted way. Adversarial attacks, on the other hand, entail adding small, carefully crafted perturbations to the input data to deceive the model and cause it to make incorrect predictions [31], [32].

And adversarial attacks can occur in federated learning when a malicious participant intentionally sends adversarial examples to the central server in an attempt to

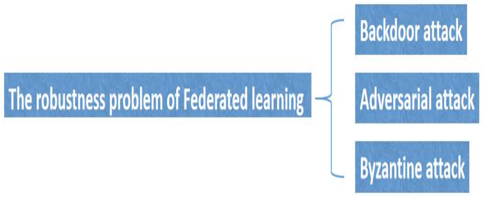

Fig. 2. The robust threat to federal learning

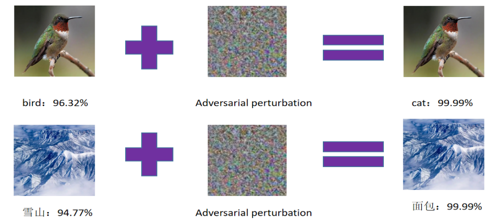

Fig. 4. Adversarial Attack

bias the model towards their own interests. This can be particularly problematic in applications such as personalized advertising or credit scoring, where the malicious participant may be motivated to gain an unfair advantage. Finally, Byzantine attacks involve one or more malicious participants in the federated learning process sending incorrect or misleading updates to the central server to disrupt the training process [35].

While federated learning can be vulnerable to certain types of attacks, there are techniques and approaches that can be used to improve the robustness and security of the process. It is important to carefully consider these issues when designing and implementing federated learning systems [38], [39]. For instance, knowledge distillation is a technique that can mitigate backdoor attacks by training a smaller [36], distilled model using the output of the original model as the target labels. This can help remove any backdoor triggers that may have been added to the original model, as the smaller model won't be able to identify them. Another technique to mitigate backdoor attacks is model erasure [44], where the model is trained to ignore specific input patterns that may be associated with the backdoor.

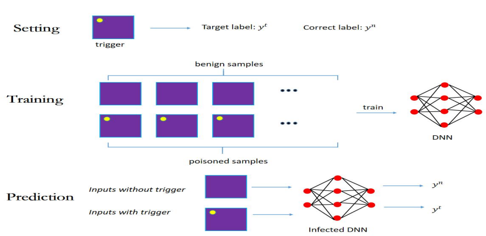

Fig. 3. Backdoor Attack

Adversarial training is a technique that involves explicitly training the model to resist adversarial examples by adding adversarial perturbations to the training data [31], [32]. This can improve the model's ability to detect and resist adversarial attacks in federated learning settings. Clustering can be used to identify malicious clients in federated learning systems subject to Byzantine attacks [35], [36]. The idea is to group participating clients based on the similarity of their updates, and to identify any clients whose updates are significantly different from the others. These clients can then be excluded from the training process, or their updates can be treated with greater suspicion to minimize the impact of their malicious behavior [37]. This paper provides an overview

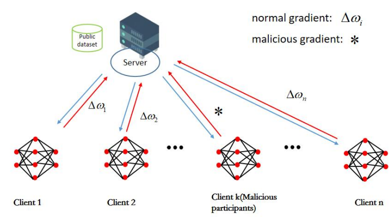

Fig. 5. Byzantine Attack

of methods to increase the robustness of federated learning models, with the aim of enhancing the credibility and security of federated learning. While previous work has addressed the security of federated learning [39], [40], [41], [42], [43], it has primarily focused on privacy leakage or backdoor attacks, with relatively few studies and reports on adversarial attacks. Building on prior work, this paper summarizes the attacks and defense methods of adversarial, backdoor, and Byzantine attacks in federated learning. A new classification method is proposed, supplementing the deficiencies of previous work on adversarial attacks. Moreover, this paper investigates a multi-level defense system against these attacks, and identifies open problems and future research directions for improving the robustness of federated learning.

II. Thraet Model

Prior to delving into the details of the threats to federal learning, it is essential to establish the connections between these threats based on different criteria. Specifically, we can categorize these threats into two main stages: the training phase and the inference phase. Additionally, we can differentiate between untargeted attacks and targeted attacks based on whether a specific target is present or not. [38], [39], [40], [41].

A. Training Phase and Inference Phase

1) Training Phase: Attacks that occur during the model training process are intended to either disrupt or impact the federated learning model itself. Backdoors are inserted into the model during the training phase to influence the resulting model outcomes [45], [46]. On the other hand, Byzantine attacks disrupt the convergence of the model by utilizing malicious clients or servers [29].

2) Inference Phase: Attacks that occur during the reasoning phase are typically intended to alter the model's reasoning outcomes and deceive it into generating incorrect outputs [47]. During the training stage, backdoor attacks involve the insertion of a backdoor into the model, whereas input deception models with triggers are utilized during the reasoning stage to cause the model to generate incorrect results. Adversarial attacks, on the other hand, leverage the model's vulnerability to disturbances and utilize samples with adversarial perturbations as input to the model, causing it to produce erroneous outcomes.

B. Untargeted and Targeted

1) Untargeted attack: Untargeted attacks are designed to compromise the integrity of the target model in an arbitrary manner. Byzantine attack is one form of an untargeted attack that involves uploading malicious gradients to the server in an arbitrary manner, with the goal of causing the global model to fail [48], [49], [50], [51].

2) Targeted Attack: A targeted attack is executed with the aim of inducing the model to produce the target label specified by the adversary for specific testing examples, while keeping the testing error for other testing examples unaffected [51].

III. Backdoor Attack

A backdoor attack on deep neural networks entails surreptitiously implanting a malicious backdoor within the model. This enables the model to function normally when processing benign inputs, but triggers a pre-defined malicious behavior when presented with a specific malicious trigger. The first neural backdoor in centralized settings can be traced back to 2013 [52], [53].

Due to the unique nature of federated learning, whereby the model is trained on individual clients, it is more susceptible to backdoor attacks compared to the general centralized training model. These types of attacks in federated learning can be divided into two categories based on the different stages at which the adversary inserts the backdoor into the training pipeline: data poisoning attacks and model poisoning attacks.

A. Data Poisoning Attack

In data poisoning attacks, it is assumed that the adversary has full control over the training data collection process of compromised clients. Then the poisoned dataset typically consists of a combination of clean data with ground-truth labels and data with backdoor triggers that have targeted labels.

1) Invisible Poisoning: In this context, the term "invisible" denotes the user's ability to execute a backdoor attack on a sample without requiring any additional actions to be performed on the sample itself.

Label flipping is a widely recognized attack in centralized machine learning (ML), as demonstrated in previous research studies [54], [55]. In addition, it is also a suitable method for the federated learning (FL) scenario, given its adversarial goal and capabilities [56].As shown in the picture, some of the samples labeled "dog" are flipped to "bird" and the samples of "cat" are flipped to "bread".

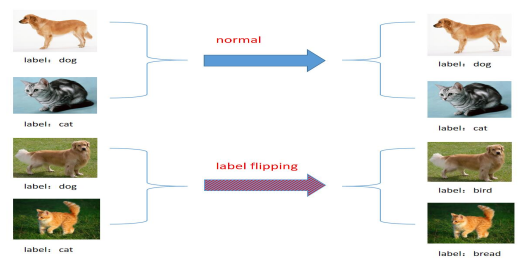

Fig. 6. Label Flipping

Nevertheless, the label flipping technique has its limitations since it necessitates modifying the label of the sample, making it less practical. Thus, an attack technique that is more covert and can deceive manual inspection would be more appealing in this scenario.A clean-label attack preserves the label of the poisoned data, and the manipulated image still appears to be a benign sample [57], [58]. This type of attack leverages feature collision, where the crafted poison examples continue to resemble the source instance in visual appearance, while being closer to the targeted instance in the latent feature space.

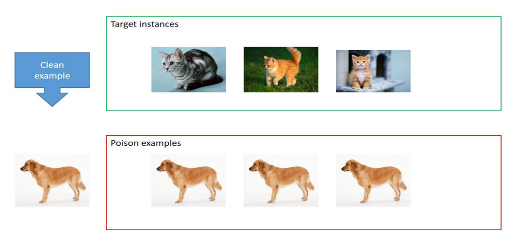

Fig. 7. Clean Label Attack

2) Visible Poisoning: Early methods of backdoor attacks in federated learning can rely on a single trigger, meaning that all corrupted clients inject the same trigger into their local training dataset. The triggers employed in this method are usually predetermined, such as a square located at redundant pixels in the image. During reasoning, the inserted trigger is used on a malicious client to activate the aggregation model [24], [27]. While the effectiveness of inserting the backdoor has been shown to be significant, the aforementioned approach merely trans-

fers backdoor attacks from centralized learning directly to federated learning, without fully leveraging the distributed nature of the latter. This is because the same trigger is embedded in all adversarial clients.

Xie et al. [59] proposed a distributed backdoor attack (DBA) that decomposes a global t rigger pattern, similar to a centralized attack, into local patterns and embeds them into different malicious clients. Compared to traditional methods that insert the same trigger, DBA is more efficient and covert due to its hidden local trigger mode, making it easier to bypass robust aggregation rules.

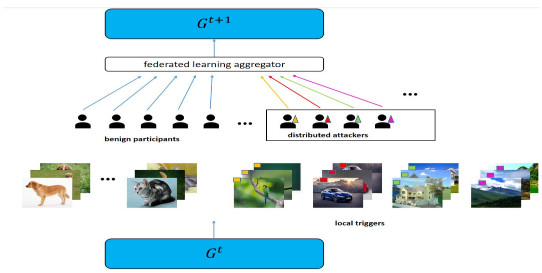

Fig. 8. Distributed Backdoor Attack

In order to enhance the effectiveness of data poisoning attacks, dynamic triggers have been proposed and applied. Salem et al [60] conducted a clear and systematic study on the feasibility of dynamic flip-flops, which can facilitate the backdoor attacks by generating antagonistic network algorithms to create triggers. This way, the same tags can be hijacked with different trigger patterns that share similar potential representations and positions. Et al [61] also conducted a direct investigation of dynamic triggers. These triggers can be flexibly produced during the physical attack phase, as they maintain their effectiveness even under substantial changes. The results of the study show that dynamically triggered backdoor attacks are more powerful, and they require new techniques to be defeated because they break the static trigger hypothesis of most current defense systems.

Despite the potential threat of data poisoning attacks in federated learning, they face many practical limitations due to the unique distributed characteristics of this approach [25]. This is because the data distribution and model aggregation steps in federated learning tend to neutralize most of the contributions of the backdoor model, leading to rapid forgetting of the backdoor by the global model. In light of this situation, Wang et al. [25] proposed selecting poisoning samples from edge data to reduce the forgetting effect caused by model updating. Dai et al. [64] proposed a new backdoor attack called Champeleon, which enables attackers to create more persistent visual backdoors by adapting to peer-to-peer images. The durability of the backdoor largely depends on the existence of two types of identical benign images that are closely related to the toxic image: 1) interferers, which are images that share the same original tag as the toxic image, and 2) facilitators, which are images with the target back tag. Interferers can cause update conflicts between toxic updates and benign updates, which may reduce the accuracy of the backdoor. Conversely, facilitators can help reintroduce backdoor information into the federated learning model and mitigate the catastrophic forgetting effect after the attacker leaves the training process. Inspired by these observations, Champeleon is designed to amplify these effects and enhance the durability of the backdoor.

B. Model Poisoning Attack

To address the limitations of data poisoning attacks, we can not only focus on improving the data poisoning technology itself, but also explore the potential of model poisoning techniques. Since the average method is the most widely-used approach for aggregating local updates from clients, a simple way to amplify the backdoor effect is to prioritize updates from adversarial clients over those from benign clients.

Bagdasaryan et al. [24] proposed the first backdoor attack against federated learning. Their approach involves training a backdoor model that closely resembles the global model, which is then used to replace the latest global model. To improve the effectiveness of this replacement, they slow down the learning rate to extend the lifespan of the backdoor model, and add an anomaly detection term to the loss function to avoid detection. This strategy requires careful evaluation of the global parameters and performs better when the global model is close to convergence.

Zhou et al. [63] proposed an optimization-based model poisoning attack that involves injecting adversarial neurons into the redundant space of a neural network to maintain the attack's concealment and persistence. To identify the redundant space, the Hessian matrix is used to measure the update distance and direction of each neuron's main task (i.e. "importance"). An additional term is then added to the loss function to prevent poisoned neurons from being injected in locations that are particularly relevant to the main task. In a similar vein, Zhang et al. [62] proposed a persistent backdoor attack called Neurotoxin. This method relies on the empirical observation that the norm of a stochastic gradient is primarily concentrated in a small number of "heavy hitter" coordinates. Neurotoxin identifies these heavy hitters using the top-k heuristic and avoids them. By avoiding directions that are most likely to receive large updates from benign devices, the chance of the backdoor being erased is mitigated.

Sun et al. [65]proposed a distance-aware attack (ADA), which enhances poisoning attacks by identifying optimized target classes in the feature space. They addressed the challenge of limited prior knowledge of customer data that competitors may face. To overcome this problem, ADA infers the pairwise distances between different categories in the potential feature space from the shared model parameters using backward error analysis. They conducted an extensive empirical evaluation of ADA by varying attack frequency in three different image classification tasks. As a result, ADA successfully improved the attack performance by 1.8 times in the most challenging cases with attack frequency of 0.01x.

C. Summary Of Federated Backdoor Attack

The primary challenge of backdoor attacks in federated learning is how to leverage the distributed nature of this approach, evade detector checks, and ensure the persistence of the backdoors during multiple rounds of update iterations. We identify three research directions for federated backdoor attacks: using multiple malicious clients to insert backdoor attacks, combining generation technology with backdoor attacks, and conducting research on persistent backdoor attacks in federated learning.

IV. Defenses against Backdoor Attack

To mitigate the problem of backdoor attacks in federated learning, various defensive techniques have been proposed. Given that we previously categorized backdoor attacks as data poisoning attacks and model poisoning attacks, we will now discuss defensive strategies for each of these attack types.

A. Defense Against Data Poisoning

The simplest approach is to filter out poisoned data samples, which aims to remove the poisoned samples from the training dataset. After the filtering process, only benign samples or purified poisoned samples are used during the training process, thus eliminating the creation of a backdoor from the source.

Tran et al. [67] were the first to investigate methods for filtering out malicious samples from the training set. They demonstrated that poisoned samples tend to leave detectable traces within the covariance range of the feature representation. Exploiting this insight, it is possible to filter out poisoned samples from the training set.Zeng et al. [68] revealed that poisoned samples of existing attacks had some high-frequency artifacts even if their trigger patterns are invisible in the input space. Based on this observation, they designed a simple yet effective filtering method based on those artifacts. Similarly, Chen et al. [69]proposed a two-stage filtering approach. In the first stage, activation values of samples in each class are clustered into two groups, and in the second stage, it is determined which clusters correspond to poisoned samples. As is well known, backdoors are triggered during the inference stage. Filtering out malicious samples from the testing samples during the inference stage can prevent the backdoor from being activated. Gao et al. [71] proposed to filter attacked samples by overlaying various image patterns on suspicious samples. The smaller the randomness of the input perturbation prediction, the higher the probability that the suspicious sample is attacked. In [72], Subedar et al. used model uncertainty to distinguish between benign and attacked samples. Later, Du et al. [73] regarded filtering as outlier detection and proposed a differential privacy-based filtering method. Recently, [74] proposed a lightweight method that can filter attacked samples or prior hypotheses of trigger patterns without labeled samples. However, Tang et al. [70] demonstrated that simple target contamination can result in malicious and benign samples being indistinguishable in the feature representation space. Therefore, most sample filtering techniques are ineffective.

V. Byzantine Attack

VI. Defenses against Byzantine Attack

VII. Adversarial Attack

Both adversarial attacks and backdoor attacks are techniques used to modify benign testing samples in order to make models misbehave during the inference process. While adversarial perturbations are sample-agnostic in universal adversarial attacks, these attacks can seem similar to backdoor attacks. Consequently, researchers who are not familiar with backdoor attacks may question their significance, as these attacks require additional controls on the training process to some extent. However, despite certain similarities, these attacks have essential differences. Firstly, adversarial attackers need to control the inference process (to a certain extent) but not the training process of models. They need to query the model results or even gradients multiple times to generate adversarial perturbations by optimization given a fixed targeted model. On the other hand, backdoor attackers require modifying some training stages (e.g., data collection, model training) without any additional requirements in the inference process. Secondly, from the perspective of attacked samples, backdoor attackers use known (i.e., non-optimized) perturbations, whereas adversarial attackers need to obtain them through the optimization process based on the output of the model. This optimization in adversarial attacks requires multiple queries, making them unable to be real-time in many cases. Finally, the mechanisms of these attacks are also essentially different. Adversarial vulnerability results from the differences in behaviors of models and humans, while backdoor attackers utilize the excessive learning ability of deep neural networks (DNNs) to build a latent connection between trigger patterns and target labels. Recently, there have been some works studying the latent connection between adversarial attacks and backdoor attacks. For example, Weng et al. [66]empirically demonstrated that defending against adversarial attacks via adversarial training may increase the risks of backdoor attacks.

VIII. Defenses against Adversarial Attack

IX. Hrbrid Defenses

X. Advanced Research and Problems

XI. Conclusion

References

[1] M. Goddard, The EU General Data Protection Regulation (GDPR): European regulation that has a global impact, International Journal of Market Research 59 (2017) 703–705.

[2] O. Gómez-Carmona, D. Casado-Mansilla, F. A. Kraemer, D. L. de Ipiña, J. GarcíaZubia, Exploring the computational cost of machine learning at the edge for human-centric internet of things, Future Generation Computer Systems 112 (2020) 670–683.

[3] J. Zhang, D. Tao, Empowering things with intelligence: A survey of the progress, challenges, and opportunities in artificial intelligence of things, IEEE Internet of Things Journal 8 (2021) 7789–7817.

[4] C. Ma, J. Koneˇcný, M. Jaggi, V. Smith, M. Jordan, P. Richtárik, M. Takáˇc, Distributed optimization with arbitrary local solvers, Optimization Methods and Software 32 (2017) 813–848.

[5] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, volume 54 of Proceedings of Machine Learning Research, pages 1273–1282. PMLR, 20–22 Apr 2017.

[6] GlobalFederatedLearningMarketbyApplication (Drug Discovery, Industrial IoT, Risk Management), Vertical (Healthcare Life Sciences, BFSI, Manufacturing, Automotive Transportation, Energy Utilities), and Region - Forecast to 2028, ”researchandmarkets.com”, Accessed date: May 12, 2023.

[7] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen, H. Yu, Federated Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning, 2019.

[8] M. Househ, E. Borycki, and A. Kushniruk. Multiple Perspectives on Artificial Intelligence in Healthcare. Springer, 2021.

[9] R. Rau, R. Wardrop, and L. Zingales. The Palgrave Handbook of Technological Finance. Springer, 2021.

[10] Fang, Xiuwen, and Mang Ye. ”Robust federated learning with noisy and heterogeneous clients.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

[11] Tang, Zhenheng, et al. ”Virtual homogeneity learning: Defending against data heterogeneity in federated learning.” International Conference on Machine Learning. PMLR, 2022.

[12] Kairouz, Peter, et al. ”Advances and open problems in federated learning.” Foundations and Trends® in Machine Learning 14.1–2 (2021): 1-210.

[13] Yang, Qiang, et al. ”Federated machine learning: Concept and applications.” ACM Transactions on Intelligent Systems and Technology (TIST) 10.2 (2019): 1-19.

[14] Huang, Wei, et al. ”Fairness and accuracy in horizontal federated learning.” Information Sciences 589 (2022): 170-185.

[15] Liu, Yang, et al. ”Vertical federated learning.” arXiv preprint arXiv:2211.12814 (2022).

[16] Ghosh, Avishek, et al. ”An efficient framework for clustered federated learning.” Advances in Neural Information Processing Systems 33 (2020): 19586-19597.

[17] Briggs, Christopher, Zhong Fan, and Peter Andras. ”Federated learning with hierarchical clustering of local updates to improve training on non-IID data.” 2020 International Joint Conference on Neural Networks (IJCNN). IEEE, 2020.

[18] Hsu, Tzu-Ming Harry, Hang Qi, and Matthew Brown. ”Federated visual classification with real-world data distribution.” Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16. Springer International Publishing, 2020.

[19] Wahab, Omar Abdel, et al. ”Federated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking systems.” IEEE Communications Surveys and Tutorials 23.2 (2021): 1342-1397.

[20] Yang, Shengwen, et al. ”Parallel distributed logistic regression for vertical federated learning without third-party coordinator.” arXiv preprint arXiv:1911.09824 (2019).

[21] Deng, Yuyang, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. ”Distributionally robust federated averaging.” Advances in neural information processing systems 33 (2020): 15111-15122.

[22] Sun, Tao, Dongsheng Li, and Bao Wang. ”Decentralized federated averaging.” IEEE Transactions on Pattern Analysis and Machine Intelligence 45.4 (2022): 4289-4301.

[23] Fallah, Alireza, Aryan Mokhtari, and Asuman Ozdaglar. ”Personalized federated learning: A meta-learning approach.” arXiv preprint arXiv:2002.07948 (2020).

[24] Bagdasaryan, Eugene, et al. ”How to backdoor federated learning.” International conference on artificial intelligence and statistics. PMLR, 2020.

[25] Wang, Hongyi, et al. ”Attack of the tails: Yes, you really can backdoor federated learning.” Advances in Neural Information Processing Systems 33 (2020): 16070-16084.

[26] Gong, Xueluan, et al. ”Backdoor attacks and defenses in federated learning: State-of-the-art, taxonomy, and future directions.” IEEE Wireless Communications (2022).

[27] Sun, Ziteng, et al. ”Can you really backdoor federated learning?.” arXiv preprint arXiv:1911.07963 (2019).

[28] Ozdayi, Mustafa Safa, Murat Kantarcioglu, and Yulia R. Gel. ”Defending against backdoors in federated learning with robust learning rate.” Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 35. No. 10. 2021.

[29] Fang, Minghong, et al. ”Local model poisoning attacks to Byzantine-Robust federated learning.” 29th USENIX security symposium (USENIX Security 20). 2020.

[30] Guo, Shangwei, et al. Byzantine-Resilient Decentralized Stochastic Gradient Descent.

[31] Zizzo, Giulio, et al. ”Fat: Federated adversarial training.” arXiv preprint arXiv:2012.01791 (2020).

[32] Chen, Chen, et al. ”Calfat: Calibrated federated adversarial training with label skewness.” Advances in Neural Information Processing Systems 35 (2022): 3569-3581.

[33] Li, Xiaoxiao, Zhao Song, and Jiaming Yang. ”Federated adversarial learning: A framework with convergence analysis.” International Conference on Machine Learning. PMLR, 2023.

[34] Zhang, Jie, et al. ”Delving into the adversarial robustness of federated learning.” arXiv preprint arXiv:2302.09479 (2023).

[35] Prakash, Saurav, and Amir Salman Avestimehr. ”Mitigating byzantine attacks in federated learning.” arXiv preprint arXiv:2010.07541 (2020).

[36] Huang, Hanxun, et al. ”Distilling Cognitive Backdoor Patterns within an Image.” arXiv preprint arXiv:2301.10908 (2023).

[37] Blanchard, Peva, et al. ”Machine learning with adversaries: Byzantine tolerant gradient descent.” Advances in neural information processing systems 30 (2017).

[38] Lyu, Lingjuan, et al. ”Privacy and robustness in federated learning: Attacks and defenses.” IEEE transactions on neural networks and learning systems (2022).

[39] Guo, Shangwei, et al. ”Robust and privacy-preserving collaborative learning: A comprehensive survey.” arXiv preprint arXiv:2112.10183 (2021).

[40] Enthoven, David, and Zaid Al-Ars. ”An overview of federated deep learning privacy attacks and defensive strategies.” Federated Learning Systems: Towards Next-Generation AI (2021): 173-196.

[41] Rodríguez-Barroso, Nuria, et al. ”Survey on federated learning threats: Concepts, taxonomy on attacks and defences, experimental study and challenges.” Information Fusion 90 (2023): 148-173.

[42] Tariq, Asadullah, et al. ”Trustworthy Federated Learning: A Survey.” arXiv preprint arXiv:2305.11537 (2023).

[43] Zhang, Yifei, et al. ”A Survey of Trustworthy Federated Learning with Perspectives on Security, Robustness, and Privacy.” arXiv preprint arXiv:2302.10637 (2023).

[44] Dal Fabbro, Nicolò, Aritra Mitra, and George J. Pappas. ”Federated TD Learning over Finite-Rate Erasure Channels: Linear Speedup under Markovian Sampling.” IEEE Control Systems Letters (2023).

[45] Miao, Chenglin, et al. ”Towards data poisoning attacks in crowd sensing systems.” Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing. 2018.

[46] Zhang, Hengtong, et al. ”Data poisoning attack against knowledge graph embedding.” arXiv preprint arXiv:1904.12052 (2019).

[47] Barreno, Marco, et al. ”Can machine learning be secure?.” Proceedings of the 2006 ACM Symposium on Information, computer and communications security. 2006.

[48] Lamport, Leslie, Robert Shostak, and Marshall Pease. ”The Byzantine generals problem.” Concurrency: the works of leslie lamport. 2019. 203-226.

[49] Xie, Cong, Oluwasanmi Koyejo, and Indranil Gupta. ”Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation.” Uncertainty in Artificial Intelligence. PMLR, 2020.

[50] Bernstein, Jeremy, et al. ”signSGD with majority vote is communication efficient and fault tolerant.” arXiv preprint arXiv:1810.05291 (2018).

[51] Damaskinos, Georgios, et al. ”Aggregathor: Byzantine machine learning via robust gradient aggregation.” Proceedings of Machine Learning and Systems 1 (2019): 81-106.

[52] Geigel, Arturo. ”Neural network trojan.” Journal of Computer Security 21.2 (2013): 191-232.

[53] Gu, Tianyu, Brendan Dolan-Gavitt, and Siddharth Garg. ”Badnets: Identifying vulnerabilities in the machine learning model supply chain.” arXiv preprint arXiv:1708.06733 (2017).

[54] Shen, Shiqi, Shruti Tople, and Prateek Saxena. ”Auror: Defending against poisoning attacks in collaborative deep learning systems.” Proceedings of the 32nd Annual Conference on Computer Security Applications. 2016.

[55] Steinhardt, Jacob, Pang Wei W. Koh, and Percy S. Liang. ”Certified defenses for data poisoning attacks.” Advances in neural information processing systems 30 (2017).

[56] Tolpegin, Vale, et al. ”Data poisoning attacks against federated learning systems.” 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 International Publishing, 2020.

[57] Shafahi, Ali, et al. ”Poison frogs! targeted clean-label poisoning attacks on neural networks.” Advances in neural information processing systems 31 (2018).

[58] Zhu, Chen, et al. ”Transferable clean-label poisoning attacks on deep neural nets.” International Conference on Machine Learning. PMLR, 2019.

[59] Xie, Chulin, et al. “DBA: Distributed Backdoor Attacks against Federated Learning.” International Conference on Learning Representations,International Conference on Learning Representations, Apr. 2020.

[60] Salem, Ahmed, et al. ”Dynamic backdoor attacks against machine learning models.” 2022 IEEE 7th European Symposium on Security and Privacy (EuroSP). IEEE, 2022.

[61] Li, Yiming, et al. “Rethinking the Trigger of Backdoor Attack.” arXiv: Cryptography and Security,arXiv: Cryptography and Security, May 2021.

[62] Zhang, Zhengming, et al. ”Neurotoxin: Durable backdoors in federated learning.” International Conference on Machine Learning. PMLR, 2022.

[63] Zhou, Xingchen, et al. ”Deep model poisoning attack on federated learning.” Future Internet 13.3 (2021): 73.

[64] Dai, Yanbo, and Songze Li. ”Chameleon: Adapting to Peer Images for Planting Durable Backdoors in Federated Learning.” arXiv preprint arXiv:2304.12961 (2023).

[65] Sun, Yuwei, Hideya Ochiai, and Jun Sakuma. ”Semi-targeted model poisoning attack on federated learning via backward error analysis.” 2022 International Joint Conference on Neural Networks (IJCNN). IEEE, 2022.

[66] Weng, Cheng-Hsin, Yan-Ting Lee, and Shan-Hung Brandon Wu. ”On the trade-off between adversarial and backdoor robustness.” Advances in Neural Information Processing Systems 33 (2020): 11973-11983.

[67] Tran, Brandon, Jerry Li, and Aleksander Madry. ”Spectral signatures in backdoor attacks.” Advances in neural information processing systems 31 (2018).

[68] Zeng, Yi, et al. ”Rethinking the backdoor attacks’ triggers: A frequency perspective.” Proceedings of the IEEE/CVF international conference on computer vision. 2021.

[69] Chen, Bryant, et al. ”Detecting backdoor attacks on deep neural networks by activation clustering.” arXiv preprint arXiv:1811.03728 (2018).

[70] Tang, Di, et al. ”Demon in the variant: Statistical analysis of DNNs for robust backdoor contamination detection.” 30th USENIX Security Symposium (USENIX Security 21). 2021.

[71] Gao, Yansong, et al. “STRIP: A Defence Against Trojan Attacks on Deep Neural Networks.” Cornell University - arXiv,Cornell University - arXiv, Feb. 2019.

[72] M. Subedar, N. Ahuja, R. Krishnan, I. J. Ndiour, and O. Tickoo, ”Deep probabilistic models to detect data poisoning attacks,” in NeurIPS Workshop, 2019.

[73] M. Du, R. Jia, and D. Song, ”Robust anomaly detection and backdoor attack detection via differential privacy,” in ICLR, 2020.

[74] M. Javaheripi, M. Samragh, G. Fields, T. Javidi, and F. Koushanfar, ”Cleann: Accelerated trojan shield for embedded neural networks,” inICCAD, 2020.