How To Backdoor Federated Learning

Authors: Eugene Bagdasaryan Cornell Tech, Cornell University
Andreas Veit Cornell Tech, Cornell University
Yiqing Hua Cornell Tech, Cornell Universityu
Deborah Estrin Cornell Tech, Cornell University
Vitaly Shmatikov Cornell Tech, Cornell University
email : first_name@cs.cornell.edu

Report Submitted by: Mohit Mohandas(20173) Data Science & Engineering,
Agamdeep Singh(20021) Data Science & Engineering,
Aayushman (20004) Data Science & Engineering

Abstract

This report provides an in-depth analysis of the research paper "How To Backdoor Federated Learning" by Eugene Bagdasaryan et al. The paper investigates the vulnerability of federated learning systems to backdoor attacks and presents a novel, more powerful backdoor attack specifically designed for federated learning. The report discusses the background, problem statement, proposed attack, evaluation methodology, experimental results, and implications of this research.

Introduction:

Federated learning is a distributed machine learning approach that allows multiple clients to collaboratively train a shared model without exposing their individual data. While federated learning offers privacy advantages, it is susceptible to various security threats. The paper by Eugene Bagdasaryan et al. focuses on the vulnerability of federated learning systems to backdoor attacks and proposes a novel, more powerful backdoor attack specifically designed for federated learning settings.

1 Background and Related Work:

Federated learning involves training a global model across multiple clients while keeping their individual data local. Clients compute local model updates and send them to a central server, which aggregates these updates to improve the global model. This process repeats iteratively until the global model converges.

Backdoor attacks are a type of poisoning attack where an attacker injects malicious behavior into a machine learning model during training. The model then exhibits this behavior when encountering specific inputs, called triggers, during inference. Backdoor attacks in federated learning are particularly challenging to detect and mitigate because the attacker can hide within a large number of clients, making it difficult to trace the source of the attack.

2 Problem Formulation:

The paper aims to address the following research question: How can an attacker perform a backdoor attack in a federated learning setting, and what are the potential countermeasures to defend against such attacks?

The authors investigate the vulnerability of federated learning systems to backdoor attacks, focusing on the unique challenges and opportunities that arise due to the distributed nature of federated learning. They propose a novel backdoor attack that takes advantage of the federated learning setting and evaluate its effectiveness against various defense mechanisms.

3 Proposed Attack

The authors propose a new backdoor attack, The paper "How To Backdoor Federated Learning" proposes a methodology for attacking federated learning, a distributed machine learning framework where multiple parties collaboratively train a model without sharing their data with each other. The proposed attack involves a malicious aggregator (i.e., a server that collects model updates from the participating parties) that adds a backdoor to the model without the knowledge or consent of the parties. This backdoor allows the attacker to control the model's behavior at inference time by adding a specific trigger pattern to the input data. The authors demonstrate the effectiveness of their attack on image classification and natural language processing tasks. The attack algorithm consists of the following steps:

1. The algorithm is used by an attacker to create a model that does not look anomalous and replace the global model after averaging with other participants' models in federated learning.
2. The input data is split into a local dataset and a backdoor dataset.
3. The attacker initializes a model X and a loss function l.
4. The attacker updates the model X by computing the gradient of the loss function on each batch of the local dataset, with a learning rate determined by the adversarial step size.
5. The attacker scales up the final model before submission to make it indistinguishable from the original model.

Algorithm 1 Model Replacement

function CONSTRAIN-AND-SCALE(D_{local}, $D_{backdoor}$)
  Initialize attacker's model X and loss function l:
  $X \leftarrow G_t$,
  $\ell \leftarrow \alpha \cdot L_{class} + (1 - \alpha) \cdot L_{ano}$
  for epoch $e \in E_{adv}$ **do**
    if $L_{class}(X, D_{backdoor}) < \epsilon$ **then**
      Early stop if model converges
      break
    end if
    for batch $b \in D_{local}$ **do**
      $b \leftarrow$ REPLACE$(c, b, D_{backdoor})$
      $X \leftarrow X - lr_{adv} \cdot \nabla \ell(X, b)$
    end for
    if $e \in step\_sched$ **then**
      $lr_{adv} \leftarrow lr_{adv}/$STEP_RATE
    end if
  end for
  Scale up the model before submission.
  $\widetilde{L}_{t+1} \leftarrow \gamma(X - G_t) + G_t$
  return $\widetilde{L}_{t+1}$

6. The model replacement is done by substituting the new global model G_{t+1} with a malicious model X in Eq. 1:

$$X = G_t + \frac{\eta}{n} \sum_{i=1}^{n} (L_{t+1}^i - G_t) \tag{1}$$

Because of the non-i.i.d. training data, each local model may be far from the current global model. As the global model converges, these deviations start to cancel out, i.e., $\sum_{i=1}^{m-1}(L_{t+1}^i - G_t) \approx 0$. Therefore, the attacker can solve for the model it needs to submit as follows:

$$\widetilde{L}_{t+1} = \gamma(X - G_t) + G_t \tag{2}$$

where $\gamma = n/\eta$ and m is the number of local models.

7. The attacker aims to minimize the loss function by updating the model X with gradients calculated on the backdoor dataset and the local dataset.

$$L_{model} = \alpha L_{class} + (1 - \alpha) L_{ano} \tag{3}$$

Here, we modify the objective (loss) function by adding an anomaly detection term L_{ano}. The objective function is defined as a combination of the classification loss (L_{class}) and the anomaly detection loss (L_{ano}) with the weight of each loss term controlled by the hyperparameter α. The

classification loss captures the accuracy on both the main and backdoor tasks, while the anomaly detection loss accounts for any type of anomaly detection.

8. The learning rate decreases over time to allow the model to converge and stabilize.

9. The attacker stops the algorithm early if the model converges to prevent further unnecessary computation.

4 Methodology/ Experiements:

4.1 The authors evaluate the effectiveness of their proposed attack and the robustness of various defense mechanisms using a series of experiments. The evaluation methodology includes:

1. Datasets: The authors use two benchmark datasets, CIFAR-10 and MNIST, to train and evaluate the backdoor attacks and defenses. These datasets contain images of handwritten digits and natural objects, respectively.

2. Attack scenarios: The authors consider various attack scenarios with different numbers of attackers and attack frequencies to assess the performance of the proposed Sign-flipping Attack and the effectiveness of defense mechanisms.

3. Defense mechanisms: The authors evaluate the robustness of several defense mechanisms against their proposed attack, including:

   (a) Model replacement: A defense where the central server replaces the model weights with the median of the weights submitted by clients. This aims to eliminate the influence of extreme weight updates that could potentially be malicious.

   (b) Gradient clipping: A defense that involves clipping the gradients of the model updates to limit their influence on the global model. This technique aims to mitigate the impact of malicious updates while preserving the contributions of benign updates.

4. Evaluation metrics: The authors use the following metrics to evaluate the performance of the attacks and defenses:

   (a) Attack success rate: The percentage of cases where the global model exhibits the backdoor behavior when encountering triggers during inference.

   (b) Clean accuracy: The classification accuracy of the global model on clean data, i.e., data without triggers.

5 Experimental Results

The experimental results demonstrate the effectiveness of the proposed Sign-flipping Attack and highlight the limitations of existing defense mechanisms. Key findings include:

1. Attack success rate: The proposed Sign-flipping Attack achieves a high success rate, causing the global model to exhibit the backdoor behavior in the majority of cases when encountering triggers during inference.
2. Clean accuracy: The attack has minimal impact on the global model's clean accuracy, indicating that the backdoor does not degrade the model's performance on clean data.
3. Model replacement defense: The model replacement defense is found to be ineffective against the Sign-flipping Attack. The attack success rate remains high even when the model replacement defense is employed.
4. Gradient clipping defense: Gradient clipping offers partial protection against the Sign-flipping Attack. While the attack success rate is reduced, the defense does not completely eliminate the backdoor.

6 Threat Model

The authors consider a threat model where an attacker aims to compromise the federated learning system by injecting a backdoor into the global model. The attacker can control a certain number of clients and their local model updates. However, the attacker cannot directly manipulate the global model or other clients' local updates. Additionally, the attacker does not have access to the central server's aggregation algorithm or any information about benign clients' data and updates.

7 Defenses

The authors examine the following defense mechanisms to mitigate the impact of backdoor attacks in federated learning:

7.1 Model Replacement

Model replacement is a defense strategy in which the central server replaces the model weights with the median of the weights submitted by clients. This aims to eliminate the influence of extreme weight updates that could potentially be malicious. However, as shown in the experimental results, model replacement is not effective against the Sign-flipping Attack, as the attack success rate remains high even when this defense is employed.

7.2 Gradient Clipping

Gradient clipping is a defense technique that involves clipping the gradients of the model updates to limit their influence on the global model. By constraining the range of gradient values, this method aims to mitigate the impact of malicious updates while preserving the contributions of benign updates. The experimental results indicate that gradient clipping offers partial protection against the Sign-flipping Attack, reducing the attack success rate but not completely eliminating the backdoor.

8 Additional Potential Defenses

Although the paper focuses on model replacement and gradient clipping, other potential defenses could be considered to address backdoor attacks in federated learning:

8.1 Federated Learning with Secure Aggregation

Secure aggregation is a cryptographic technique that allows the central server to aggregate clients' local model updates without learning the individual updates. By concealing the updates, secure aggregation can help protect against eavesdropping and tampering attacks. However, its effectiveness against backdoor attacks remains to be investigated.

8.2 Outlier Detection

Outlier detection methods can be employed to identify and remove clients that submit anomalous model updates, which could potentially be malicious. These methods may include statistical tests, clustering algorithms, or machine learning-based approaches. The challenge lies in accurately distinguishing malicious updates from benign ones, especially when the attacker employs sophisticated techniques like gradient masking.

9 Implications

The research paper "How To Backdoor Federated Learning" by Eugene Bagdasaryan et al. provides valuable insights into the vulnerability of federated learning systems to backdoor attacks and the limitations of existing defense mechanisms. The proposed Sign-flipping Attack demonstrates that attackers can exploit the unique characteristics of federated learning to inject backdoors into the global model, while the evaluation of defenses highlights the need for more robust countermeasures.

The findings of this paper have important implications for the security of federated learning systems, emphasizing the need for the development of new

defense mechanisms that can effectively protect against backdoor attacks. Additionally, the results contribute to a better understanding of the challenges associated with securing distributed machine learning systems and can inform the design of more secure federated learning protocols.

10 Improvements

Our goal with the improvements was to increase persistence at the possible cost of the trade-off in accuracy, for which we tried two approaches.

10.1 Approach 1: scaling (X - G) further to increase persistence

The standard replacement follows the following procedure:

$$\widetilde{L}_{t+1} = \gamma(X - G_t) + G_t \tag{4}$$

Where λ is solved for model replacement. We hypothesised that scaling λ further than model replacement will take the model along the line joining X and G further up. As the federated learning bring the model back towards G, it will pass through X hence the region near X will be elongated, making the model persist longer.

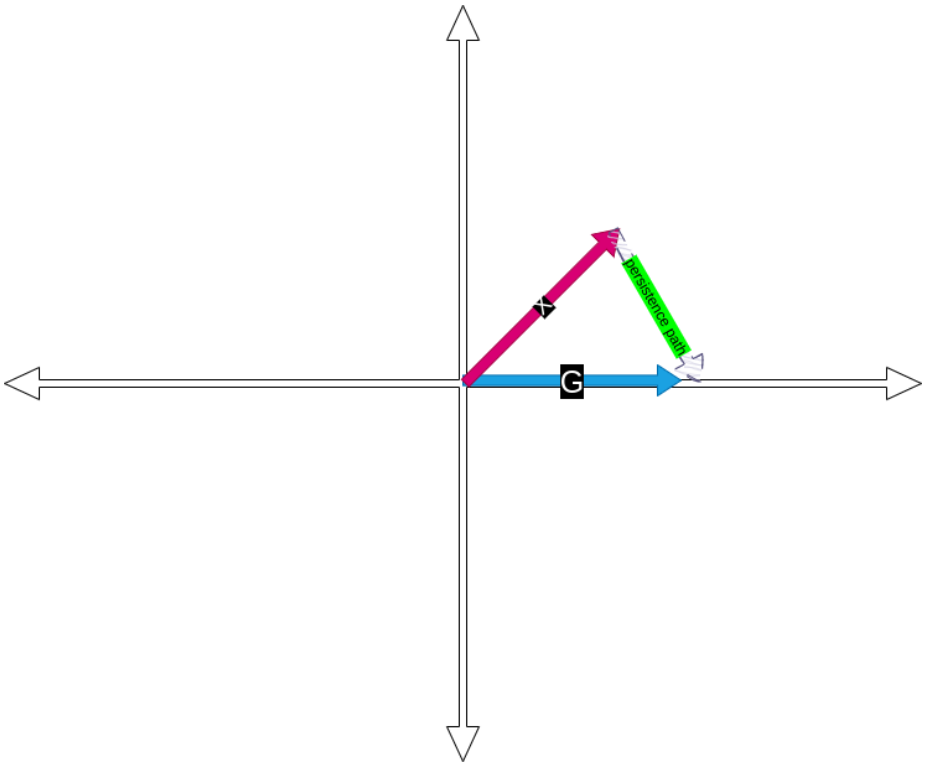

Figure 1: Old replacement model with shorter persistence length

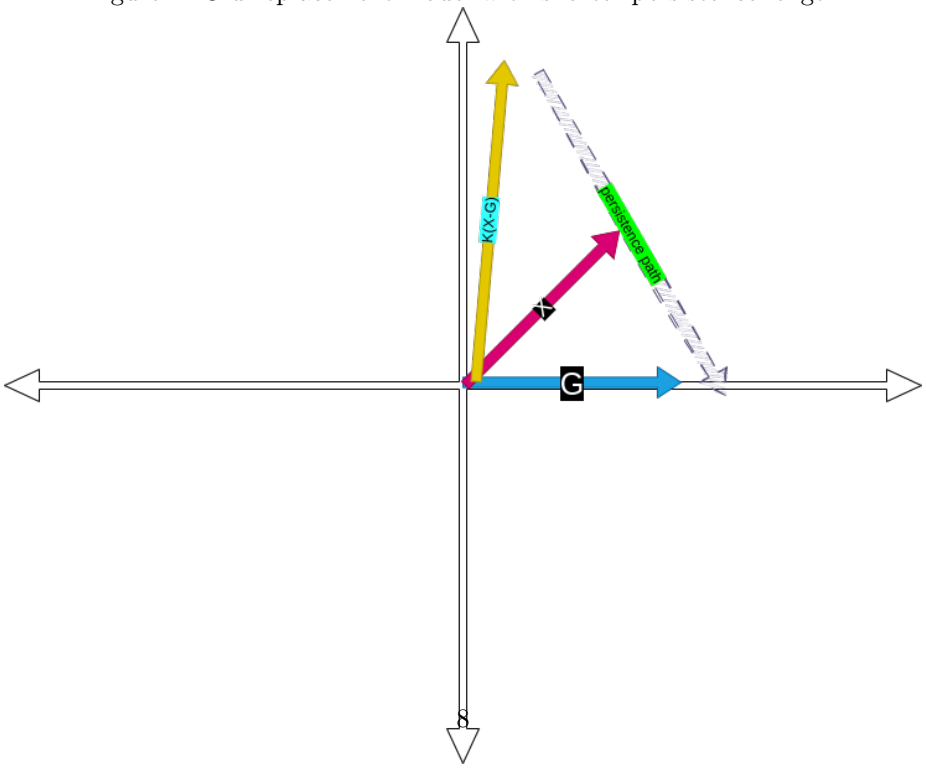

Figure 2: New replacement model with longer persistence length

10.2 Approach 2: Change weightage

We tried the following update scheme:

$$\widetilde{L}_{t+1} = \gamma(X(1+\beta) - G_t(1-\beta)) + G_t \tag{5}$$

This model changes the weightage of the replacement. Keeping the model closer to G while inputting controllable portions of X.
We hypothesised the gradients in the approach would be smaller as we are closer to G than before but still get good accuracy due to X.

10.3 Results

Unfortunately, both methods failed. Another method would be to approach this from the PoV of BNNs as they allow more change while giving robust results. The current model was too sensitive to change and di not give us accuracy.