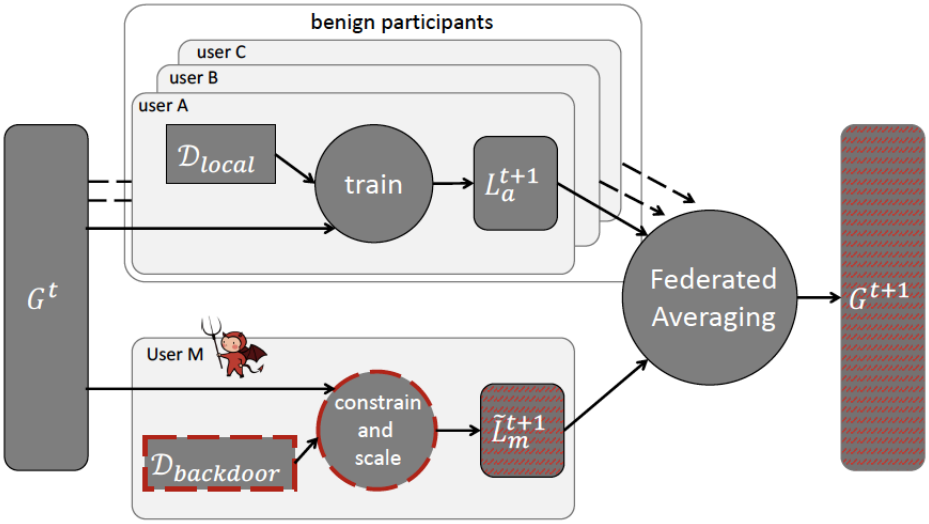
2 How To Backdoor Federated Learning

[2]

presented by: Gokberk Yar

2.1 Problem Statement

Federated machine learning is a framework in which multiple machines participate in the learning process where the data is not shared between the machines and the final model is created by aggregating the model parameters of participating machines/models. This paper introduces a backdoor attack on Federated learning. Here an attacker introduces a backdoor data into a small subset of the machines and when the parameters from these machines are finally aggregated into the main model the model gets compromised.
We see the general architecture of federated learning in below figure. User A,B and C are good machines and User M is malicious withe backdoor data and all the updates from these machines are aggregated in the central server.



2.2 Threat Model

In this attack the attacker has full access into one or several participant machines. Which means that the attacker controls the local training data, the training procedure and hyper parameters such has learning rate. In essence the attacker can modify the model parameters before they are sent to the central server for aggregation.

2.3 Methodology

Naive approach Incorporate the backdoor poisoning from Badnets paper [1], where the attacker simply adds the backdoor examples into the training process.
Model replacement The above naive approach does not work in federated learning as the contribution of majority of the models gets cancelled out before reaching the final model. In order to preserve the contribution of the few malicious machines the paper suggests and model replacement approach which substitutes the new global model G^t with X which multiples the contribution of malicious machines by scaling factor $\frac{\eta}{n}$

$$X = G^t + \frac{\eta}{n} \sum_{i=1}^{m} (L_i^{t+1} - G^t) \qquad (2)$$

In the above equation (2), G^t is the weights of global model and L^t is the weights of local model. if the attacker does not know η and n then the scaling factor can be approximated by $\gamma < \frac{\eta}{n}$. This is a single shot attack.

Constrain and Scale Latest proposals of Federated learning incorporates secure aggregation which has no way of telling if the weights from the malicious machines are good or bad but if secure aggregation if not present then the central server might filter anomalous weights. So in order for the method to work in either of the cases the model has to maintain high accuracy on main task and the backdoor task. So a generic objective is suggested below.

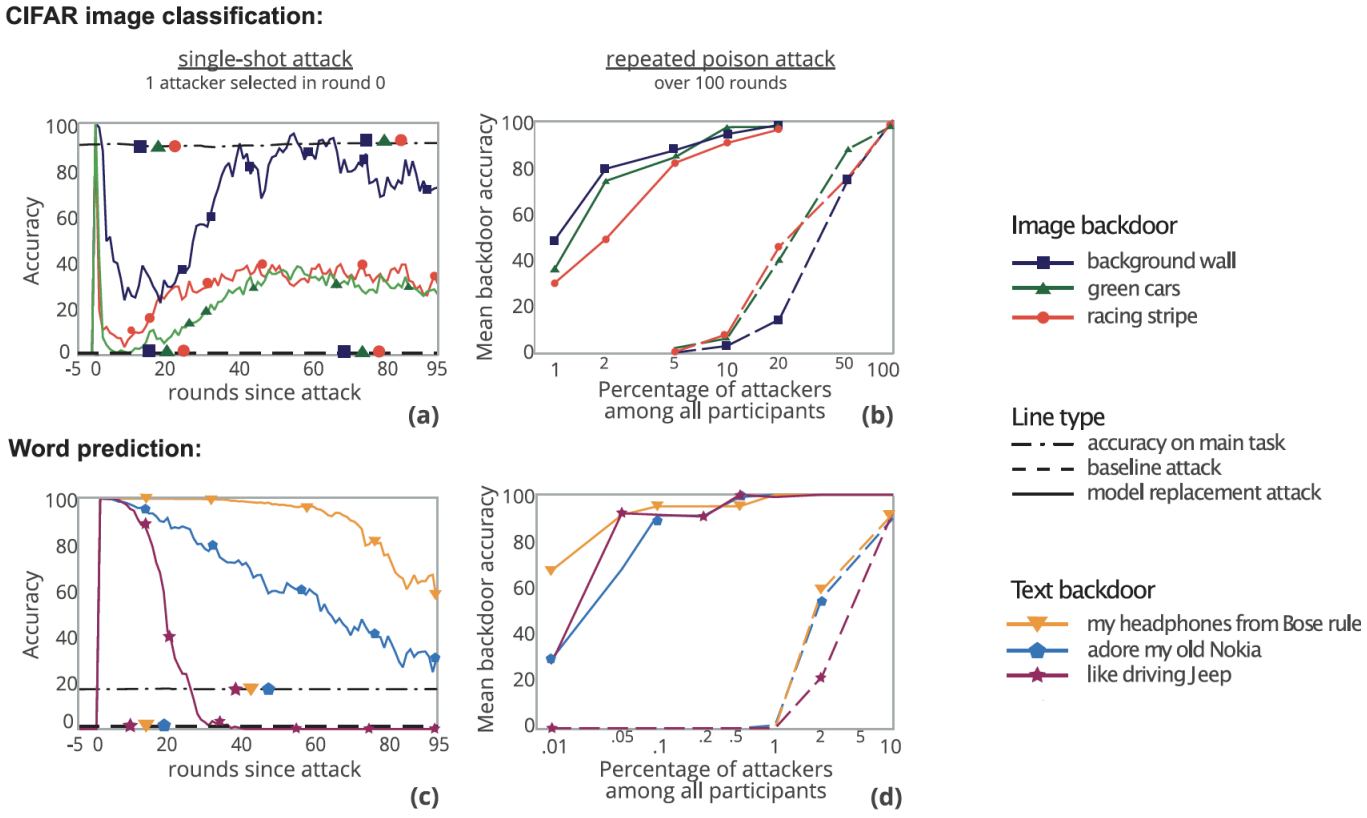
$$\mathcal{L}_{model} = \alpha \mathcal{L}_{class} + (1 - \alpha)\mathcal{L}_{ano} \tag{4}$$

Here the first loss L_{class} makes sure that the model does well on both the main task and the backdoor task. L_{ano} accounts for any anomaly detection such as p-norm distance between the weight matrices.
Train and Scale For anomaly detectors which only look at the magnitude of weights, the attacker can wait till the model converges and then introduce the malicious model weights scaled by γ such that the value is permitted by a bound S like below.

$$\gamma = \frac{S}{||X - G^t||_2} \tag{5}$$

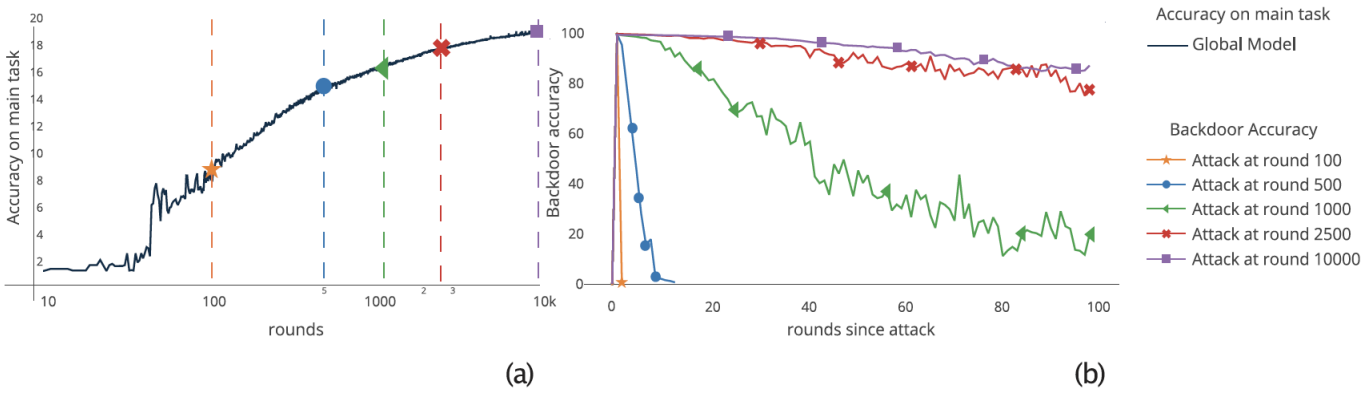
2.4 Results

The attack procedure on federated learning is tested on two tasks. Image classification and word prediction. The backdoor introduced in training process of malicious machines is called as semantic backdoor. Semantic backdoor do not require modification of input at inference time. For example in the image classification task the backdoor can be unusual color car images such as green color.



In single shot attack only a single malicious participant is selected in a round poisoning. The baseline is not successful in reducing Model accuracy as explained in the methodology section. In image classification the accuracy reduces immediately after the backdoor is introduced and then starts to rise which could due be to different learning rates and non-convex objective. For word prediction the success depends on the type of semantic backdoor.

In repeated attack more than one participant is selected. Here we can see more the participants higher the backdoor accuracy.



It is possible to introduce the backdoor at various stages of training rounds. In the above figure we see, backdoors introduced in earlier rounds are forgotten quickly compared to backdoors that are introduced later.

2.5 Class discussion

What are the memory requirement for the local model? We do not expect the local model to be very large due to compute limitations and hence expect certain optimizations to make it work.
When there are multiple participants do they submit same weights to the global model? It is not clear from the paper how this situation is handled but we do expect some randomization to handle this.
Why doesn't the central server catch gradient updates that are too far? The central server algorithm is only aggregating the gradients and hence does not clip or alter the updates.
Why does the accuracy drastically reduce in the graph for image classification? This accuracy drops when the backdoor is introduced and increases due to possible reasons mentioned in paper.
The main difference between the naive approach and the model replacement is the scaling factor
Is anomaly detection good for Model poisoning? because the data is non iid and so the model updates are meant to far of This is true and it is not of the possible limitation.
The technique works better when the model is close to convergence because at that time the malicious model updates will have greater effect.

References

[1] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. "BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain". In: *CoRR* abs/1708.06733 (2017). arXiv: 1708.06733. URL: http://arxiv.org/abs/1708.06733.

[2] Eugene Bagdasaryan et al. "How To Backdoor Federated Learning". In: *CoRR* abs/1807.00459 (2018). arXiv: 1807.00459. URL: http://arxiv.org/abs/1807.00459.

[3] Zuxuan Wu et al. "Making an Invisibility Cloak: Real World Adversarial Attacks on Object Detectors". In: *CoRR* abs/1910.14667 (2019). arXiv: 1910.14667. URL: http://arxiv.org/abs/1910.14667.