FLock: Defending Malicious Behaviors in Federated Learning with Blockchain

Nanqing Dong*
Department of Computer Science
University of Oxford
nanqing.dong@cs.ox.ac.uk

Jiahao Sun
FLock.io
sun@flock.io

Zhipeng Wang
Department of Computing
Imperial College London
zhipeng.wang20@imperial.ac.uk

Shuoying Zhang
FLock.io
shuoying@flock.io

Shuhao Zheng
School of Computer Science
McGill University
shuhao.zheng@mail.mcgill.ca

Abstract

Federated learning (FL) is a promising way to allow multiple data owners (clients) to collaboratively train machine learning models without compromising data privacy. Yet, existing FL solutions usually rely on a centralized aggregator for model weight aggregation, while assuming clients are honest. Even if data privacy can still be preserved, the problem of single-point failure and data poisoning attack from malicious clients remains unresolved. To tackle this challenge, we propose to use distributed ledger technology (DLT) to achieve **FLock**, a secure and reliable decentralized **F**ederated **L**earning system built on bl**ock**chain. To guarantee model quality, we design a novel peer-to-peer (P2P) review and reward/slash mechanism to detect and deter malicious clients, powered by on-chain smart contracts. The reward/slash mechanism, in addition, serves as incentives for participants to honestly upload and review model parameters in the FLock system. FLock thus improves the performance and the robustness of FL systems in a fully P2P manner.

1 Introduction

Federated learning (FL) [5] is a machine learning (ML) paradigm where data owners/clients collectively train a ML model under the orchestration of a central server, while the clients' raw data stay local – they are not transferred or shared with any other party. Most FL implementations follow a centralised, 'hub-and-spoke' topology. The aggregation server distributes global models for clients to train, collects their model updates, then re-distribute updated global model, until training is complete. [3]

Since FL does not involve collecting data from all data services and storing it on a server, it is seen as a promising solution to conduct machine learning while protecting user privacy. Privacy-preserving machine learning has become increasingly valuable due to regulatory restrictions and customer demand. Globally, over 70% of countries have introduced regulations on data privacy; in particular, GDPR [2] in Europe and HIPAA [7] in the US have strict restrictions on what companies can access and store. This has either increased compliance and operations costs, or resulted in companies discontinuing related services [6]. With FL, companies can analyse data without centralising them first, resulting in simpler compliance

*Authors are arranged in alphabetical order.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

procedures and fewer regulatory restrictions. On the other hand, consumers face the dilemma between data protection and functionality. According to a 2021 survey [4], while two thirds of the customers do not have a positive view on companies' data protection policies, under 20% are happy to share personal information to get additional value such as better services – the rest either do not want to share or consider it as a compromise. FL will bring a much stronger user proposition, one that makes data collection obsolete and does not ask users to choose between privacy and functionality.

In conventional FL settings, clients submit local model updates to the central server, which stores and aggregates local updates, before sharing global model updates with the clients back. The central server could become a single point of failure for both the security and the performance of the system, prone to model poisoning, privacy leakage, network delay, and targeted delays.

Blockchain [9], which acts as an immutable and decentralised ledger, is well-placed to tackle the issue of the server being a central point of failure. There have been several proposed approaches replacing a central server in with a blockchain [8].

A drawback is that these approaches assume that clients do not intend to undermine the global model and can commit resources to staying online and local training. However, there are several ways malicious clients could impair the global model performance. For instance, they could drop out from training, refuse to upload model updates, train models using low-quality or fake data, or just spam the system with random model updates.

The lack of systematic protection against malicious clients is a limiting factor to FL's widespread and decentralised adoption, in conventional and on-chain settings. Implementation is thus limited to existing trust circles, where rules of collaboration are more enforceable. To pave the way for large-scale cooperation, we need an automated and enforceable mechanism to ensure execution integrity among participants and protect against malicious attempts.

We aim to combine blockchain's consensus mechanisms with a decentralised scoring mechanism to preserve performance as well as decentralisation. There is no trusted party to authorise which transactions are valid or organise the participants on the blockchain; instead, any party has the opportunity to propose new blocks and the participants are organised by a pre-agreed definition of the canonical record (chain of transactions with the most computation done on it) and an incentive scheme for building on the canonical record (mining rewards). On top of the blockchain, smart contracts allow the execution of arbitrary code or rules, forming a decentralised and distributed machine with shared states among each participating party.

We propose a secure and reliable decentralised Federated Learning system built on blockchain ("FLock") to reduce the trust in server/other clients required. In addition to on-chain smart contracts replacing the central server in the model aggregation process, there are two main contributions: a reward/slashing mechanism to ensure model performance, and a P2P model evaluation system. Smart contracts provide transparency and ensure there are no side channel attacks from the central server; the reward/slashing mechanism deters data poisoning attacks; and finally, the P2P model evaluation system safeguards model performance and filters out malicious clients in the multi-round setting.

2 System Design

The maliciously secure FLock system consists of two phases: namely the **Setup** phase and the **Training** phase. Participants prepare the data and stake tokens during the Setup phase and collaboratively train the model in an auditable way during multiple Training phases, with each Training phase consisting of four steps: **local training**, **on-chain aggregation**, **committee voting**, and **reward/slash**.

2.1 Participants

There are mainly three kinds of participants in FLock system (cf. Figure 1):

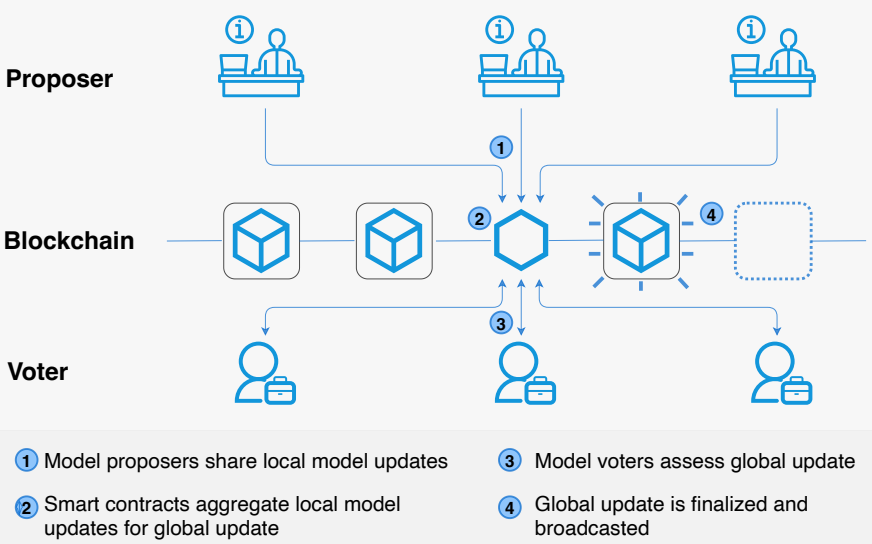

Figure 1: System design of Flock FL.

- *Proposer*: Blockchain nodes selected to train the model on their local private dataset and report weight updates.
- *Voter*: Blockchain nodes selected to form a committee to audit the aggregated model weights during each Training phase.
- *Miner*: Blockchain nodes that actively order transactions on the FLock blockchain and produce blocks.

In this work, we mainly focus on describing the unique working procedure for clients and voters, while the miner's part is similar to that in other blockchain systems. It is worth noting that one person in the real world is not restricted to acting as only one participant in the FLock system, which adds extra security requirements to the whole system since the malicious strategy becomes more complex in this scenario. The mechanism design of FLock system also targets to defend against such complex malicious behaviors to achieve a practical system ready to use.

2.2 System Pipeline

2.2.1 Setup Phase

To participate in a federated learning task, any participant should first stake some tokens into the FLock system as collaterals to represent their commitment. The staked tokens will later be used in the reward/slash phase to reward honest actors and punish malicious actors. For miners, staked tokens also contribute to the probability to be selected as the block producer and receive block rewards, known as the Proof-of-Stake (PoS) mechanism.

In order to participate in model training, a proposer should prepare a private training dataset and optionally a validation dataset for local model selection. Any node who wants to become a committee voter should prepare a local test dataset for global model weight auditing. Obviously, any proposer can leave out part of their training dataset as the test dataset and participate as a voter. A novel reward/slash mechanism is applied in the FLock system to incentive honest proposers and voters and punish malicious players, which is explained in Section

2.2.2 Training Phase

One training phase consists of the following four steps:

1. *Local training*: Each node receives the same global model weights at the end of each training phase. Then, at the beginning of a new training phase, several nodes are randomly selected as model proposers for this phase from all the nodes eligible to train models. Only the selected nodes are permitted to locally train their model based on the global model and propose the local model updates to the FLock system.
2. *On-chain aggregation*: After receiving the local model weight updates from the proposers in this round, the miner selected to produce the block will aggregate the weights via the FedAvg protocol and publish the global model weights on the blockchain. However, as the miner might be malicious and censor specific proposers to gain benefits, it is crucial for other miners to validate the correctness of model aggregation. Therefore, after the global model weights are proposed, other miners will start a voting round for the validity of the aggregation results before entering the next step.
3. *Committee voting*: To avoid bad model weight updates and punish malicious proposers, an auditing committee is randomly selected among all the eligible participants to evaluate the global model weights. Each committee voter locally evaluates the model performance on the previous prepared test dataset and calculates a voting score based on the performace. Afterwards, each voter simply reports the voting score to the blockchain miners for tallying the votes. An automatic on-chain protocol will determine if the global model weights in this round will be discarded according to the voting scores.
4. *Reward/Slash*: To incentivize honest behaviors and get rid of bad actors, we re-distribute the staked tokens according to the voting scores at the end of each training phase. For the proposers, the reward/slash is given based on the voting scores. For the voters, we calculate the amount to reward/slash based on the difference between their individual voting score and the final aggregation score and the voting direction. We configure the system such that eventually honest participants will get rewards and malicious participants will get slashed. Since each participant should stake enough tokens to participate, malicious participants will eventually quit the system as their tokens get slashed. In this way, the FLock system facilitates a secure and truthful federated learning process without sacrificing the final model performance.

2.3 Threat Model

The malicious behaviors we consider in designing the system include:

- *Malicious Proposer*: A proposer may use bad or duplicate data for training, or report bad model weight updates to harm the global model performance. She may also decline to upload her model weight updates or be unresponsive.
- *Malicious Voter*: A voter may vote extremely in the reverse direction to maximally bias the aggregation result and gain benefits. She may also decline to vote or be unresponsive.

Since one person in the real world can register for multiple nodes in the FLock system, it is also necessary to consider collusion behaviors among different roles in the system. In this work, we theoretically prove that the system parameters can be configured to defend against all the possible malicious behaviors under the honest majority assumption, which is generally believed to be true in most PoS blockchain systems. It is worth noting that defending against malicious miners is not considered in this work, which already has effective solutions such as FlashBots [1].

3 Theoretical Analysis

In this section, we provide theoretical analysis on expected returns for proposers and voters, and calculate the optimal system parameters to incentivise good behaviours and penalize malicious participants.

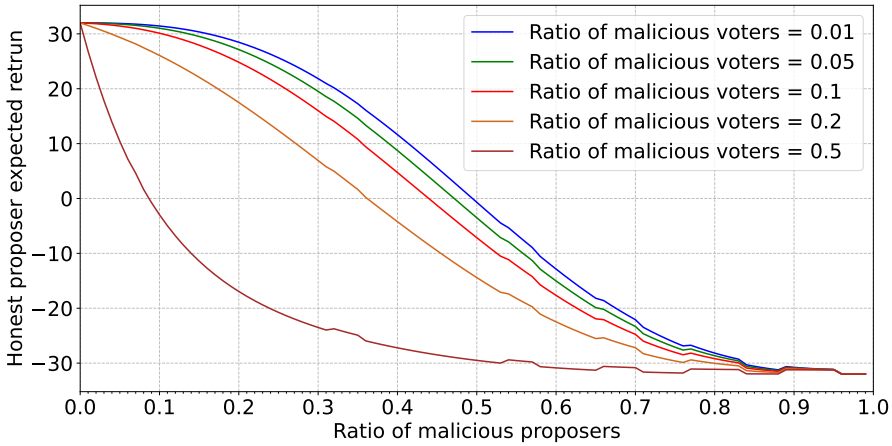

Figure 2: The expected rewards of honest proposers over the ratio of malicious proposers.

3.1 Expected Return for Participants

The expected rewards for clients who participate in proposing are shown in Equation 1. Note that N is the number of participants, N_p is the number of proposers per round, N_v is the number of voters per round, l_p is the ratio of malicious proposers and l_v is the ratio of malicious voters. α and β are two system parameters determining the participants' rewards and penalties. T is the threshold number configured by the system.

$$\mathbb{E}(R^p) = f(\alpha, \beta, T, N, N_p, N_v, l_p, l_v) \tag{1}$$

As shown in Figure 2, the expected rewards of honest proposers decrease over the ratio of malicious proposers given fixed system parameters. Our theoretical analysis results prove that there are optimal system parameters (i.e., α, β and T) to incentivise honest participants and penalize malicious participants.

Acknowledgements

This work was supported and funded by FLock.io LTD.

References

[1] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash Boys 2.0: Frontrunning, Transaction Reordering, and Consensus Instability in Decentralized Exchanges. *IEEE Symposium on Security and Privacy (SP)*, 2020.

[2] European Commission. General data protection regulation, 2016.

[3] Mu Li, David G Andersen, Alexander J Smola, and Kai Yu. Communication efficient distributed machine learning with the parameter server. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[4] Brodherson Marc, Broitman Adam, Cherok Jason, and Robinson Kelsey. A customer-centric approach to marketing in a privacy-first world. `https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/a-customer-centric-approach-to-marketing-in-a-privacy-first-world`, 2021.

[5] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[6] UNCTAD. Data Protection and Privacy Legislation Worldwide. https://unctad.org/page/data-protection-and-privacy-legislation-worldwide, 2022.

[7] US Department of Health and Human Services. Health insurance portability and accountability act, 2017.

[8] Zhilin Wang and Qin Hu. Blockchain-based federated learning: A comprehensive survey. *arXiv preprint arXiv:2110.02182*, 2021.

[9] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151:1–32, 2014.