A Scheme of Anti Gradient Leakage of Federated Learning Based on Blockchain

Xin Zhang[1], Yuanzhen Liu[1], Yanbo Yang[1(✉)], Jiawei Zhang[2], Teng Li[2], and Baoshan Li[1]

[1] School of Information Engineering, Inner Mongolia University of Science & Technology, Bao Tou 014010, China
lz@stu.imust.edu.cn, yangyanbo@imust.edu.cn
[2] School of Cyber Engineering, Xidian University, Xi'an 710071, China

Abstract. Federated learning provides a new solution for data security and privacy protection in the process of machine learning. In distributed learning, interactive gradient rather than direct exchange of data avoids the direct acquisition of data by malicious participants. The latest research shows that the gradient will also leak the data privacy during user training, current solutions to gradient leakage problems are mainly divided into two categories: 1. Encrypt the parameters of the model updating process by using the cryptography scheme; 2. Increase the interference noise for the gradient through thinning, gradient compression, adding noise and other schemes. However, the cryptographic scheme cannot be targeted at the aggregator, and the noise scheme has too much impact on the performance of the model. This paper uses the trust mechanism of blockchain to design an anti gradient leakage scheme for this problem: when the participants upload the gradient, the noise is added through the blockchain smart contract, and the noise value is randomly divided into multiple copies and stored in the blockchain. The central server obtains the noise sum of all participants through the smart contract, and aggregates the gradient to remove the noise. In the training process, all participants can only obtain the aggregation gradient and noise gradient, and cannot recover user data. At the same time, due to the automatic execution and tamper proof characteristics of blockchain smart contracts, malicious interference in the data exchange process is avoided.

Keywords: Federated Learning · Blockchain · Ring signature · Deep Leakage from Gradients · Gradient Update

1 Introduction

With the further development of network communication, the Internet also needs a higher level of security and stronger privacy protection. Various countries and regions have issued a series of regulations related to privacy protection, which put forward strict requirements for data security and privacy protection [1]. Machine learning needs to use a large amount of data for training, but the protection of data and privacy makes it difficult

Q. Jiang et al. (Eds.): SPNCE 2022, LNICST 496, pp. 95–108, 2023.
https://doi.org/10.1007/978-3-031-30623-5_7

for data to circulate and form data islands, which cannot release the greater value of data. Traditional machine learning based on central servers faces serious privacy and security challenges and cannot achieve ubiquitous secure AI for future networks. Furthermore, traditional centralized machine learning schemes may not be suitable for ubiquitous AI due to the huge overhead introduced by centralized data aggregation and processing [2]. As an emerging distributed machine learning scheme, federated learning provides new solutions to the privacy and security issues faced by machine learning. In federated learning, participating devices collaboratively train a shared model through their local data. Unlike traditional machine learning schemes, only model updates are uploaded to the centralized parameter server instead of the original data, providing tighter privacy for machine learning. Under federated learning, the original data is always stored locally, avoiding direct leakage of privacy [2]. However, according to the latest research, the process of model updating still faces the risk of privacy leakage. For example, malicious participants can infer user data by obtaining gradient updates during user training, which poses a great threat to data security and user privacy in federated learning [3].

With the rapid development of blockchain technology in recent years, its own "Trust" feature has created a reliable cooperation mechanism. It can provide a trusted environment in an untrustworthy network and can provide secure and trusted services in numerous scenarios, solving the problems faced by traditional centralized servers. The combined use of blockchain and distributed learning effectively improves the confidentiality of data and the security of computing and can meet a variety of different application scenarios.

We designed a blockchain-based solution for federal learning gradient leakage, using smart contracts to add noise to the gradient when users submit gradient data, while storing the noise value in the blockchain by secret sharing, and the central server removes the noise after aggregating the model, which ensures the machine learning training accuracy and solves the gradient leakage problem at the same time. The main contributions of this paper are as follows:

(1)In this paper, a federated learning improvement protocol is designed to add noise to the model to prevent gradient leakage when participants submit data and to remove noise after the central server aggregates the model to avoid affecting training accuracy.
(2)Using blockchain and smart contracts to store several copies of noise values obtained by gradient hash recalculation and a smart contract to calculate all the shared noise and return to the central server, storing the noise values shared by each participant in the blockchain, avoiding the malicious participant from directly obtaining the noise data to recover the gradient and then infer the real data of the user. The application of ring signature in the process guarantees the identity security of data uploaders.

2 Related Work

Jin et al. [4] proposed an advanced data leakage attack and theoretically demonstrated that the attack can effectively recover bulk data from shared aggregation gradients. Zhao et al. [5] proposed a simple but reliable method to extract accurate data from the gradients. Wu et al. [6] proposed a new method called gradient privacy leakage (PLFG) to

infer sensitive information through gradients only. Shafee et al. [7] point out the attacks on privacy by deep learning models and some solutions to them. Wang et al. [8] proposed a generalized gradient privacy attack called SAPAG, which attacks user privacy based on gradient differences as a distance metric; Wei et al. [9] proposed a principled framework for evaluating and comparing different forms of client-side privacy leakage attacks, showing how an adversary can reconstruct private local training data by simply analyzing shared parameter updates from local training (e.g., local gradient or weight update vectors). Ren et al. [10] describe the attack as a regression problem and optimize two branches of the generative model by minimizing the distance between gradients; Wainakh et al. [3, 11] propose using LLG to extract training data labels from user-shared gradients; Jia et al. [12] speed up image reconstruction by imposing a priori image information and improving initialization. Hu et al. [1] propose a new inference attack, called Source Inference Attack (SIA), which optimally estimates the source of training members; Huang et al. [13] evaluated the advantages of three defense mechanisms against gradient inversion attacks, showed the trade-offs of these defense methods in terms of privacy disclosure and data utility, and found that combining them in an appropriate way reduces the effectiveness of the attack; Yuan et al. [14] made the first attempt to explore record-level privacy leakage for NLP tasks in FL by exploiting the complexity of language modeling to study the exposure of records of interest in federation aggregation. Two related attacks are proposed by monitoring exposure patterns to identify the corresponding clients when extracting specific records.

Lin et al. [15] used sparsification to reduce exchange costs and achieve reasonable accuracy with various model structures. Zheng et al. added a dropout layer before feeding data to the classifier and obtained better results with an appropriate dropout rate; Wei et al. [16], for example, clarified that the traditional server-coordinated differential privacy approach is insufficient to protect the privacy of training data reasons, analyzed the privacy utility tradeoff for Fed-CDP to provide differential privacy guarantees, and proposed a dynamic attenuation noise injection strategy to further improve Fed-CDP's accuracy and resilience. Fu et al. understood the utility of differential privacy in FL by adjusting the number of iterations performed and formally derived the convergence under differential privacy noise in FL based on the FedAvg algorithm speed; Wang et al. [17] suggested reshaping and tailoring gradient methods for differentially private distributed optimization and proposed two differentially privacy-oriented gradient methods to ensure privacy and optimality; Wu et al. [18] applied local differential privacy techniques to local gradients to protect user privacy and combined randomly sampled items into pseudo-interaction items in order to protect the items with which users interact in order to achieve anonymity. Zhao et al. [19] proposed a new CDL framework, PrivateDL, which allows efficient transfer of relational knowledge from sensitive data to public data in a privacy-preserving manner and enables participants to learn local models together based on public data with noise-proof labels. PrivateDL creates a privacy gap between local models and private datasets, thus ensuring privacy from attacks launched against local models through gradient sharing. Hya et al. [20] proposed a privacy-preserving network transformation method using random permutations in software protection extensions (SGX), which protects model parameters from being inferred by curious servers and dishonest clients; Dxa et al. [21] proposed an entropy-based gradient compression

(EGC) mechanism to reduce communication overhead, where EGC selects the transmission based on the entropy of the gradient terms gradient, which can achieve a high compression ratio without sacrificing accuracy; So et al. [22] introduced a new metric to quantify the privacy assurance checkout of joint learning over multiple training sessions and to develop a structured user selection policy to ensure the long-term privacy of each user.

Current responses to the gradient leakage problem mainly have two solution ideas: 1. Use cryptographic schemes such as homomorphic encryption to encrypt the updated model parameters and then upload them to the central server, which will encrypt the updated model parameters using cryptographic schemes, and only the central server can decrypt and aggregate them into a new model, In such schemes, it is difficult for other participants to obtain the user gradient, but the central server can decrypt the user gradient parameters, thereby revealing the user privacy; 2. Add noise to the model, and when users upload the gradient parameters, add appropriate noise to the data so that the participants cannot directly access each round of training gradients, thus making it difficult to recover the data.

3 Preliminary Knowledge

3.1 Blockchain

In blockchain technology, decentralization is its most essential feature. Each node in the network backs up the complete ledger information and provides a credible transaction environment in an untrustworthy network, laying a solid foundation of "trust" and creating a reliable cooperation mechanism that has a wide range of application prospects.

The consensus mechanism contained in the blockchain enables each unrelated node to verify and confirm the data in the network, thus generating trust and reaching consensus; the cryptographic algorithm escorts the anonymity, immutability and unforgeability of the blockchain, which is the bottom line of whether a chain is trustworthy and has basic security; a smart contract is a contract defined in digital form that can automatically enforce its terms. It puts the contract in the form of code on the blockchain and executes it automatically under the agreed conditions. The immutable and traceable nature of the blockchain provides a secure and trusted environment for smart contracts to operate.

The protocols through smart contracts means that the participants can only execute according to the protocols specified in the contract, and the contract can be executed automatically when the contract trigger conditions are met, which brings into play its advantages in cost efficiency and greatly avoids the interference of malicious behaviors during the normal execution of the contract.

3.2 Federated Learning

Federated learning as an emerging distributed machine learning scheme provides a new solution to address the privacy and security issues faced by machine learning. In federated learning, the core idea is to build a global model based on virtual fused data by exchanging model parameters or intermediate results without exchanging local individual or sample

data through distributed model training among multiple data sources with local data, so as to achieve a balance between data privacy protection and data sharing computation, i.e., the new paradigm of "The New Application Paradigm" of "data available but not visible, data not moving model moving".

3.3 Ring Signature

A Ring signature is a simplified group signature that eliminates the need for signature group creation and group administrators. It provides a clever way to achieve unconditional anonymity of the signer's identity while avoiding the problem of excessive group administrator privileges. In ring signatures, the signer cannot be confirmed as the identity of the signer, and this unconditional anonymity is very useful in special environments where information needs to be protected for a long time.

In a ring signature, the attacker can also not determine which member of the ring generated the signature. The signature generated by one member of the ring can be verified by all others, while other members of the ring cannot forge the signature of the real signer. An external attacker can not forge a signature for the data even if it is based on obtaining a valid ring signature.

The unique nature of ring signatures can be widely used in anonymous electronic elections, e-government, e-money systems, key distribution, and multi-party secure computing, thus becoming a hot topic of current research.

3.4 Gradient Leaks

It has long been assumed that gradients can be shared securely, i.e., gradient exchange does not disclose training data. Instead, studies have shown that malicious participants can obtain private training data from publicly shared gradients [23].

In the training process of machine learning, the machine learning model is optimized by continuously updating the gradient to obtain the optimal model. The gradient was calculated as described in Eq. (1):

$$\nabla W_{t,i} = \frac{\partial l(F(x_{t,i}, W_t), y_{t,i})}{\partial W_t} \tag{1}$$

where W_t is the model weight parameter for round t, $x_{t,i}$, $y_{t,i}$ are the data inputs and labels for round t of participant i, $F(\bullet)$ is the predicted value of the input data under the current model parameters, and $l(\bullet)$ is the loss function constructed using the predicted values with the true labels.

Under the condition that the learning rate is η, the weights are updated in each round as described in Eq. (2):

$$W_{t+1} = W_t - \eta \overline{\nabla W_t} \tag{2}$$

Federation learning, in which the gradients computed separately for each participant need to be aggregated as described in Eq. (3).

$$\nabla W_{t+1} = \frac{1}{N} \sum_{i}^{N} \nabla W_{t,i} \tag{3}$$

To recover the data from the gradient, we first initialize a random virtual input x' and label input y'. Then, we input these "virtual data" into the model and obtain the "virtual gradient" $\nabla W'$, as described in Eq. (4):

$$\nabla W' = \frac{\partial l\left(F\left(x', W\right), y'\right)}{\partial W} \tag{4}$$

Optimizing the spurious gradients close to the original also brings the dummy data close to the real training data. Given the gradient at a particular step, we obtain the training data by minimizing the following objectives, as described in Eq. (5).

$$x', y' = \arg\min_{x', y'} \left\| \nabla W' - \nabla W \right\|^2 \tag{5}$$

The paradigm distance $||\nabla W' - \nabla W||^2$ in the objective function is differentiable, and the virtual input x' and label y' can be optimized using a standard gradient-based approach. After several iterations of optimization, the real data can be recovered (Fig. 1).

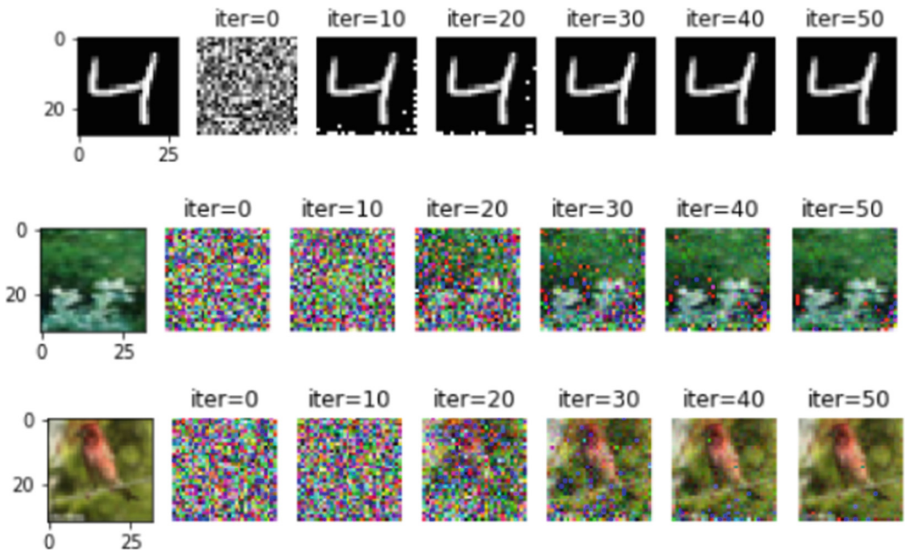

Fig. 1. Recover data from MNIST, cifar-100 and cifar-10 datasets by gradient leakage method

4 Gradient Protection Principles

As shown in Fig. 2, although adding noise to the gradient parameters of the training model can enhance the privacy protection in the machine learning process, the noise has a serious impact on the machine learning training effect; adding too little noise cannot achieve effective privacy protection, and adding too much noise causes serious degradation to the machine learning training accuracy [23].

Fig. 2. Gradient leakage data recovery with noise distributions with variance of 0.001, 0.01 and 0. 1 respectively

In the scheme we design, each participant adds noise R_t to the gradient $\nabla W_{t,i}$ updated in each round and sends it to the central server, which receives the gradient $\nabla W'_{t,i}$ asdescribed in Eq. (6):

$$\nabla W'_{t,i} = \nabla W_{t,i} + R_{t,i} \tag{6}$$

The central server aggregates the incoming gradients as described in Eq. (7):

$$\overline{\nabla W_t}' = \frac{1}{N} \sum_{i}^{N} \nabla W'_{t,i} \tag{7}$$

The participant sends the generated noise to multiple participants using secure multi-party computation $F_s(\bullet)$ as described in Eq. (8):

$$r_1, r_2, ..., r_s = F_s\left(R_{t,i}\right) \tag{8}$$

Multiple security calculations in Eq. 8 establish a security protocol. The security protocol enables each participant to obtain the corresponding information without including other information output, preventing the malicious node and other participants to know which participant the information belongs to and the privacy of the participants.

The participants jointly calculate the sum of all random numbers as described in Eq. (9).

$$R_t = \frac{1}{N} \sum_{i}^{N} R_{t,i} = F_s^{-1}(r_1, r_2 \ldots, r_s) \tag{9}$$

Central server recovery gradient as described in Eq. (10).

$$\begin{aligned}\nabla W_t &= \overline{\nabla W_t'} - R_t \\ &= \frac{1}{N}\sum_{i}^{N} \nabla W_{t,i}' - \frac{1}{N}\sum_{i}^{N} R_{t,i} \\ &= \frac{1}{N}\sum_{i}^{N} (\nabla W_{t,i}' - R_{t,i}) \\ &= \frac{1}{N}\sum_{i}^{N} (\nabla W_{t,i} + R_{t,i} - R_{t,i}) \\ &= \frac{1}{N}\sum_{i}^{N} \nabla W_{t,i}\end{aligned} \tag{10}$$

It can be seen that the data recovered by the central server is the same as the aggregated gradient without any added noise, which ensures that the training accuracy will not be affected by the added noise. By passing the added noise through the secure multi-party computation, neither any participant nor the central server can directly recover the noise value or infer the true gradient, thus ensuring the user's privacy.

5 An Overview of the System

5.1 System Model

As shown in Fig. 3, the whole system is composed of three parts: the blockchain network, participants, and central server. The details are described as follows.

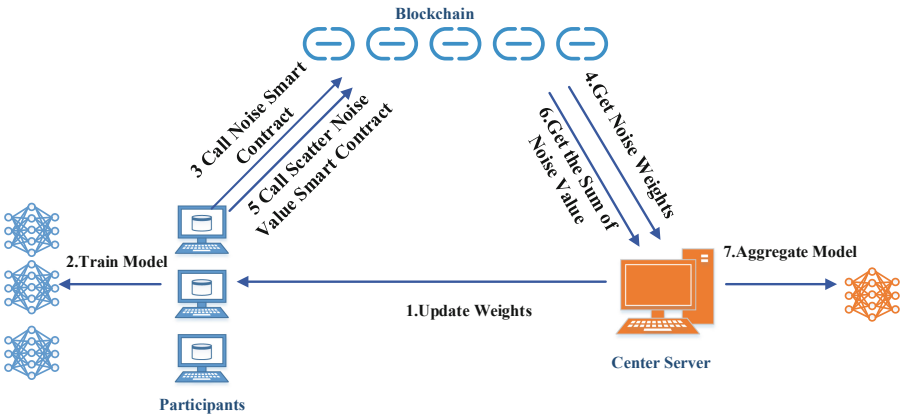

Fig. 3. System model diagram

(1)The Blockchain network, which provides a secure and trusted execution environment for the system, guarantees the reliable operation of smart contracts, stores data, and safeguards data security.
(2)The participant, the provider of the training data, trains the model parameters using the data he has.
(3)The central server, which is responsible for the aggregation of model gradients and updating of model weights.

5.2 Threat Model

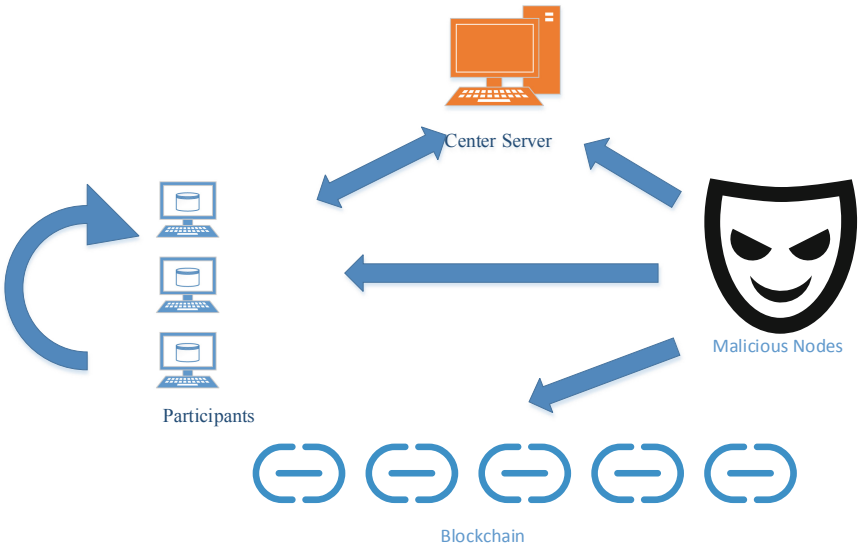

Fig. 4. Threat model diagram

Although blockchain provides a secure and trusted computing environment, it still faces the risk of attacks on user data and privacy. Figure 4 shows the threats faced by the system: Participants may send wrong data; the central server will infer user privacy based on the gradient data raised by participants; malicious nodes will disrupt the training process during the machine learning training, making the training of machine learning difficult and exposing the identity of data parties and data privacy; blockchain management users can, based on the data stored in the blockchain, recover the data of the participating parties, expose the user's identity based on the transaction records of the blockchain, etc.

5.3 Solution Principle

In the federal learning training process, the central server directly obtains the gradient parameters of the model trained by the participant to aggregate into new model parameters and can also directly recover the user data according to the gradient. In the whole

process, not only can the central server directly obtain the gradient parameters, but also the transmission process is easy to leak the gradient data, which threatens the privacy and security of users.

In summary, we designed a blockchain-based solution for federal learning gradient leakage: when the participant finishes training and uploads the gradient data, the smart contract automatically adds the noise value associated with the gradient hash and saves the gradient after adding noise in the blockchain, i.e., it is impossible to recover the user's data directly through gradient leakage. The participants sign the model after adding noise by ring signature, and any participant can not confirm who uploaded the data after uploading.

When the central server aggregates the data, the participants use the hash value of the gradient as a parameter to invoke the smart contract that shares the noise value. The smart contract first validates the uploaded gradient data before proceeding to the next step. The smart contract recalculates the noise value by the hash of the gradient and divides it into random copies, each of which is stored in the blockchain. The malicious node has no way to know the number of copies of the noise value divided, and the operation is signed by a ring signature, so each participant has no way to know to which participant the shared noise value stored in the blockchain belongs, and therefore cannot obtain the real noise value corresponding to each participant. The noise sum required by the central server in aggregating data is directly returned by the smart contract after calculating the shared noise values of all participants. Finally, the latest model parameters are calculated. The data operations are executed by smart contracts during the process, and the execution process cannot be interfered with artificially. This greatly improves the system's ability to resist malicious nodes.

5.4 Program Construction

The specific process of the scheme is as follows:

(1) **Training of models**
The participants use the model parameters shared by the central server to train the machine learning and use their own data to calculate the gradient ∇W_i.

(2) **Gradient uploads**
Participant i invokes the add-noise smart contract with the gradient ∇W_i as a parameter, and the smart contract calculates the hash $H_i = Hash(\nabla W_i)$ of ∇W_i, generates the noise value $R_i = Random(H_i)$ with H_i as a random number seed, calculates the gradient $\nabla W_i' = \nabla W_i + R_i$ of the add-noise, and stores $\nabla W_i'$ in the blockchain.

(3) **Validate the data and share the noise values**
Participant i invokes the shared noise value smart contract with the hash value H_i of the gradient ∇W_i as an argument, and the smart contract first verifies whether the uploaded noise gradient contains the noise gradient $\nabla W_i'$ corresponding to H_i: The smart contract first calculates the noise value $R_i' = Random(H_i)$ with H_i. Using the noise value R_i' and the noise gradient $\nabla W_i'$, it calculates the gradient $\nabla W_i'' = \nabla W_i' - R_i'$ and checks whether the hash value $H_i' = Hash(\nabla W_i'')$ of $\nabla W_i''$ is consistent with H_i. If it is consistent, it means that the gradient ∇W_i uploaded by

participant i has not been changed, otherwise, it means that the noise gradient stored in the blockchain does not match with the gradient uploaded by the participant.

After verifying the data, the smart contract divides $R_r{}'$ into n_i random copies, generates n_i random numbers $R_{i,1}, R_{i,2}, ..., R_{i,n_i}$, where $R_{i,1} + R_{i,2} +, ..., +R_{i,n_i} = R_i^{'}$, and stores the random number $\mathrm{R_{i,1}}, \mathrm{R_{i,2}}, ..., \mathrm{R_{i,n_i}}$ in the designated account address on the blockchain.

(4) **Obtain the noise value**

The central server aggregates the gradients by first obtaining the noise sum, which is obtained by the central server by invoking the smart contract that computes the noise sum. The smart contract sums up all the noise $R_{1,1}, R_{1,2}, \ldots, R_{1,n_1}, R_{2,1}, R_{2,2}, \ldots, R_{2,n_2}, \ldots$ in the account after division to obtain R, and returns R to the central server.

(5) **Polymerization gradient**

The central server obtains the noise gradient $\nabla W_1', \nabla W_2', \ldots, \nabla W_n'$ from the blockchain and calculates the average gradient using $\overline{\nabla W_t} = \frac{1}{n}(\sum_{i=1}^{n} \nabla W_i' - R)$. The new round model parameters are $W_{t+1} = W_t - \eta\overline{\nabla W_t}$. After sharing the new model parameters with each participant, the next round of training is performed until a compliant machine learning model is obtained.

6 Analysis of Security

The federal learning gradient leakage solution proposed in this paper effectively prevents the user privacy problem caused by gradient leakage and improves the problem of model accuracy degradation caused by adding noise. At the same time, it protects the user's identity security through blockchain smart contract and ring signature technology, and greatly avoids malicious behaviors in the training process.

As shown in Table 1, compared with the recent method, the method protects the participant information, so that other participants cannot judge which participant the information belongs to; resist malicious nodes and central servers to recover information from the gradient; resist malicious nodes to tampering with information; ensure the accuracy of model parameters and ensure the training effect.

In the gradient uploading stage, the participant invokes the smart contract to add noise to the gradient, and what is stored in the blockchain is not the real gradient. At the same time, the data uploaded by the participants is signed by the ring signature technology, so others cannot tell which user the data in the blockchain belongs to and cannot find multiple rounds of training ladders for the same participant.

In the shared noise value phase, the participants invoke the smart contract with the gradient hash as a parameter, and first verify that the noise gradient stored in the blockchain matches the uploaded gradient based on the gradient hash, and subsequently, the noise value is randomly divided into multiple copies. Since a ring signature is used for the operation, a malicious attacker cannot confirm which participant the divided noise values belong to, and all the divided noise values are stored in the same account address, making it impossible to recover the true noise values. Therefore, it is impossible

Table 1. Compares with the most recent scheme

	The malicious nodes and other participants are unable to determine which participant the information belongs to	Resisting malicious nodes from recovering information from the gradient	Resthe central server to recover information from the gradient	The accuracy of the model parameters is not affected, and the training effect is guaranteed	Verify whether the malicious node destroys the information and determine whether the participant sends the error information
The method of this paper	√	√	√	√	√
ESMFL	-	√	-	√	-
FedGCN	-	√	-	-	-
Fed-CDP	-	√	-	√	-

to recover the true gradient based on the noise gradient and the divided noise value, and thus protect the user privacy and security.

In the acquire noise sum stage, the central server obtains the noise sum directly through the smart contract, avoiding the direct acquisition of noise values and thus recovering the true gradient.

In the aggregated gradient phase, the central server uses the noise gradient and the noise sum to calculate the average gradient and update the model parameters. Compared with other add-noise schemes, there is no loss of model parameter accuracy and the machine learning training effect is guaranteed.

In the improved scheme, the ring signature ensures the identity security of the participants; the data storage process is automatically completed by the smart contract in the blockchain and cannot be controlled by humans; the results are written in the blockchain; and the malicious nodes cannot destroy the training model due to the tamper-evident nature of the blockchain. In addition, due to the introduction of the blockchain network, the transmission of information between parties does not need to find a separate transmission channel, which greatly enhances the transmission security and reduces the network communication volume.

7 Conclusion

Aiming at the problem of gradient leaking privacy in federated learning, this paper proposes a federated learning training scheme based on blockchain. In this scheme, by adding enough noise to the gradient, malicious participants and the central server can not infer user privacy by using the gradient. The central server removes the noise after aggregating the gradient uploaded by the participants, so the final aggregated gradient will not affect the original gradient, The performance degradation of the final model is

avoided. Because in the process of adding and removing noise, it is executed through blockchain, and the added noise is randomly divided, malicious participants can only obtain the aggregated model parameters, and cannot obtain the gradient parameters of each participant. Malicious participants cannot steal user privacy through gradient disclosure. Experiments show that this scheme provides a secure solution to the privacy security caused by gradient leakage in federated learning, and ensures the performance of the training model.

Acknowledgments. This research is supported by Natural Science Foundation of Inner Mongolia Autonomous Region (2020LH06006); Major science and technology projects of Inner Mongolia Autonomous Region (2019ZD025); Innovation fund of Inner Mongolia University of science and Technology (2019QDL-B51); Inner Mongolia discipline inspection and supervision big data open project (IMDBD2020021); Kundulun District Science and technology plan of Baotou City, Inner Mongolia (YF2021011).

References

1. Hu, H., Salcic, Z., Sun, L.: Source inference attacks in federated learning. In: 2021 IEEE International Conference on Data Mining (ICDM), pp. 1102–1107 (2021)
2. Jahani-Nezhad, T., Maddah-Ali, M.A., Li, S.: SwiftAgg: communication-efficient and dropout-resistant secure aggregation for federated learning with worst-case security guarantees. arXiv preprint arXiv (2022)
3. Wainakh, A., Ventola, F., Müig, T.: User label leakage from gradients in federated learning. arXiv preprint arXiv (2021)
4. Jin, X., Chen, P.Y., Hsu, C.Y.: CAFE: catastrophic data leakage in vertical federated learning. In: Advances in Neural Information Processing Systems, pp. 994–1006 (2021)
5. Zhao, B., Mopuri, K.R., Bilen, H.: iDLG: improved deep leakage from gradients. arXiv preprint arXiv (2020)
6. Wu, F.: PLFG: a privacy attack method based on gradients for federated learning. In: Yu, S., Mueller, P., Qian, J. (eds.) SPDE 2020. CCIS, vol. 1268, pp. 191–204. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-9129-7_14
7. Shafee, A., Awaad, T.A.: Privacy attacks against deep learning models and their countermeasures. J. Syst. Archit. **114**, 101940 (2020)
8. Wang, Y., Deng, J., Guo, D.: SAPAG: a self-adaptive privacy attack from gradients. arXiv preprint arXiv (2020)
9. Wei, W., Liu, L., Lope,r M.: A framework for evaluating gradient leakage attacks in federated learning. arXiv preprint arXiv (2020)
10. Ren, H., Deng, J., Xie, X.: GRNN: generative regression neural network – a data leakage attack for federated learning. ACM Trans. Intell. Syst. Technol. (TIST) **13**(4), 1–24 (2022)
11. Wainakh, A., Müig, T., Grube, T.: Label leakage from gradients in distributed machine learning. In: 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC). IEEE (2021)
12. Jia, Q., Hansen, L.K.:What can we learn from gradients? (2020)
13. Huang, Y., Gupta, S., Song, Z.: Evaluating gradient inversion attacks and defenses in federated learning. arXiv e-prints. Advances in Neural Information Processing Systems, vol. 34, pp. 7232–7241 (2021)
14. Yuan, X., Ma, X., Zhang, L.: Beyond class-level privacy leakage: breaking record-level privacy in federated learning. IEEE Internet Things J. **99** (2021)

15. Lin, S., Wang, C., Li, H.: ESMFL: efficient and secure models for federated learning. arXiv preprint arXiv (2020)
16. Wei, W., Liu, L., Wu, Y.: Gradient-leakage resilient federated learning. In: 2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS). IEEE (2021)
17. Wang, Y., Nedic, A.: Tailoring gradient methods for differentially-private distributed optimization. arXiv preprint arXiv (2022)
18. Wu, C., Wu, F., Cao, Y.: FedGNN: federated graph neural network for privacy-preserving recommendation. arXiv preprint arXiv (2021)
19. Zhao, Q., Zhao, C., Cui, S.: PrivateDL: privacy-preserving collaborative deep learning against leakage from gradient sharing. Int. J. Intell. Syst. **35**(8), 1262–1279 (2020)
20. Hya, D., Li, H., Xxa, D.: PPCL: privacy-preserving collaborative learning for mitigating indirect information leakage. Inf. Sci. **548**, 423–437 (2021)
21. Dxa, B., Yuan, M., Di, K.: EGC: entropy-based gradient compression for distributed deep learning - ScienceDirect. Inf. Sci. **548**, 118–134 (2021)
22. So, J., Ali, R.E., Guler, B.: Securing secure aggregation: mitigating multi-round privacy leakage in federated learning. arXiv preprint arXiv (2021)
23. Zhu, L., Liu, Z., Han, S.: Deep leakage from gradients. In: Advances in Neural Information Processing Systems **32** (2019)