Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8005769, 9 pages
https://doi.org/10.1155/2021/8005769

Research Article

An IoT Crossdomain Access Decision-Making Method Based on Federated Learning

Chao Li,[1] **Fan Li**,[1,2] **Zhiqiang Hao**,[3] **Lihua Yin**,[1] **Zhe Sun**,[1] **and Chonghua Wang**[3]

[1]*Cyberspace Institute of Advanced Technology, Guangzhou University, 510700, China*
[2]*Guangxi Key Laboratory of Cryptography and Information Security, 541004, China*
[3]*China Industrial Control Systems Cyber Emergency Response Team, China*

Correspondence should be addressed to Lihua Yin; yinlh@gzhu.edu.cn

Received 6 August 2021; Accepted 4 December 2021; Published 27 December 2021

Academic Editor: Lei Chen

Crossdomain collaboration allows smart devices work together in different Internet of Things (IoT) domains. Trusted third party-based solutions require to fully understand the access information of the collaboration participants to implement crossdomain access control, which brings privacy risk. In this paper, we propose a federated learning-based crossdomain access decision-making method (FCAD), which builds a crossdomain access decision-making model without sharing privacy information of collaboration participants. Crossdomain access logs are extracted to construct a training dataset. Data enhancement method is used to address the uneven distribution of the dataset. Federated learning and gradient aggregation methods are used to prevent privacy leaks. The experiments on the public dataset show that FCAD obtains a prediction accuracy of 83.6% in the existing crossdomain access system.

1. Introduction

Internet of Things (IoT) allows connections of heterogeneous smart devices. More than 25 billion smart devices will be connected through IoT by 2025 [1]. Some IoT service providers support heterogeneous devices collaboration cross their domains. For example, IFTTT provides a platform where users can defined multidevice connection rules [2]. It allows devices in multiple domains to work together by translating user rules to requests in these domains. Suppose that a user defines a rule "*if the door is open, then open the house lights*." The smart door lock is provided by Philips, and the smart lights are provided by Samsung. They are managed by different domain, which are the IoT platforms named *Philips Hue* and *SmartThings*. IFTTT sends requests to the two IoT platforms to make the rule effective. The crossdomain collaboration makes some IoT operations be more convenient.

Crossdomain access control is used to prevent unauthorized access in crossdomain collaboration. The "domain" in crossdomain collaboration means domain, and the "crossdomain collaboration" is to describe the collaboration of devices belong to different managers. Existing crossdomain access control methods often rely on a trusted third party (TTP). The TTP verifies the requests legality and makes crossdomain access decisions. For example, the National Health Information Network (NHIN) [3] unites IoT domains of multiple hospitals by providing a trusted third party platform, to form a virtual alliance of medical systems. This alliance guarantees the freedom of information flowing between doctors and patients and implements the crossdomain access control. In the meantime, many companies are providing crossplatform access services, such as SmartThings [4] and Google Home [5]. The IoT platform makes access decision and translates user rules to requests in different domains, to achieve crossdomain collaboration.

However, the trusted third party-based solutions lack secure access control policies for crossdomain collaboration. Access control policies are the rules which are used to make an *Allow* or *Deny* decision for an access request [6]. Access control policies are mainly configured by experts or generated by policy mining [7, 8]. In policy mining, logs are used as input of policy mining algorithm, to automatically mine access control policies. However, to protect user's privacy, participants are often unwilling to share access logs, which make it difficult for IoT platforms to get the access logs of collaboration participants among different domains. Then, the IoT platform can only use the incomplete information to mine access control policies. This brings many problems like credential leakage, incomplete revocation, and incorrect policy enforcement [9, 10]. The lack of secure crossdomain access control methods puts collaboration participants at risk of being attacked. A security crossdomain collaboration solution is needed that shares no access logs of participants but completes the crossdomain access decision-making.

In this paper, we propose a crossdomain access decision-making method based on federated learning to solve above problems. Federated learning allows users to train their machine learning model locally and then builds a global model by aggregating the shared local model gradients. In the crossdomain collaboration system using federated learning, local access logs will not be shared. Participants use their local access logs to train their own models. Then, gradients of models are exchanged and aggregated in multiple rounds. Finally, we will obtain a global crossdomain access decision-making model to make decisions for crossdomain access requests. Our contributions are as follows:

(i) We propose a log preprocessing method to address uneven distribution of crossdomain access logs. By using a data enhancement algorithm, the logs are transformed as the input of learning algorithms

(ii) We propose a federated learning-based crossdomain access decision-making method (FCAD), to build a crossdomain access decision-making model. The model can decide whether to allow or deny the crossdomain access requests without sharing privacy information of collaboration participants

(iii) We evaluate the effectiveness of FCAD on a public dataset. The experimental results show that FCAD can obtain a prediction accuracy of 83.6% in a crossdomain collaboration system

The rest of this paper is organized as follows. In section 2, we describe related works. The system design is given in Section 3. Section 4 shows the experiments of FCAD. Section 5 summarizes our work.

2. Related Works

We divide existing access decision-making methods into policy-based methods and learning-based methods.

2.1. Policy-Based Methods. The policy-based methods make access decisions by the access control policies, which are used to describe the system security constraint. Traditional works rely on field expert knowledge. Neumann et al. [11] use role engineering based on professional knowledge to define roles, permissions, constraints, and role hierarchies for the role-based access control (RBAC), but the limited expert knowledge causes the unstable quality of mined policies. Automatic mining access control policies from the existing access control information are the focus of most researches. Iyer et al. [8] propose an attribute-based access control (ABAC) policy generation method, which can extract positive and negative authorization rules from given access control information, and then mine policy entries. Xu et al. [12] propose an ABAC policy mining method based on access logs and attribute information. They convert access logs into user-authority mappings and iteratively obtain a policy set equivalent to the original access control system. Access control policy mining based on algorithm is also a feasible method. Carlos et al. [13] propose UNICORN, which uses the deterministic annealing and mean-field approximation to achieve a universal access control policy mining.

2.2. Learning-Based Methods. The powerful effect of deep learning on distinguishing normal and abnormal behaviors makes it widely used in IoT access decision-making [14]. Narouei et al. [15] use natural language processing (NLP) to analyze system documents, which contains security information. Access control content is identified in the documents written with natural language for access decision-making. Mocanu et al. [16] propose a neural network-based ABAC policy mining method. The method adds attribute data to the access log and converts them into a vector, thereby using them as the input of the neural network model. This method can discover the hidden distribution of the data. Karimi et al. [17] propose a policy mining method based on unsupervised learning algorithm, which mines policies from the extracted policy rule pattern. Jabal et al. [18] propose a framework for learning ABAC policies from examples and context information. The framework achieves good results in both real logs and synthetic logs. Xiang et al. [7] propose time changing decision tree to process the existing access logs. The time changing decision tree records and continuously monitors the current access control constraints of the system. When a new access request does not meet the constraints, the administrator will be notified of risks.

These works use the access control information to mine policies or train a learning model, which are designed for the access decision-making in a single system. In the crossdomain collaboration system, the access control information is protected strictly due to the privacy risk, which leads to the loss effectiveness of these methods. A new crossdomain collaboration solution is needed, which can make access decisions without sharing access information of participants. In this paper, we propose a federated learning-based crossdomain access decision-making method (FCAD). The federated learning enables the crossdomain central server to get

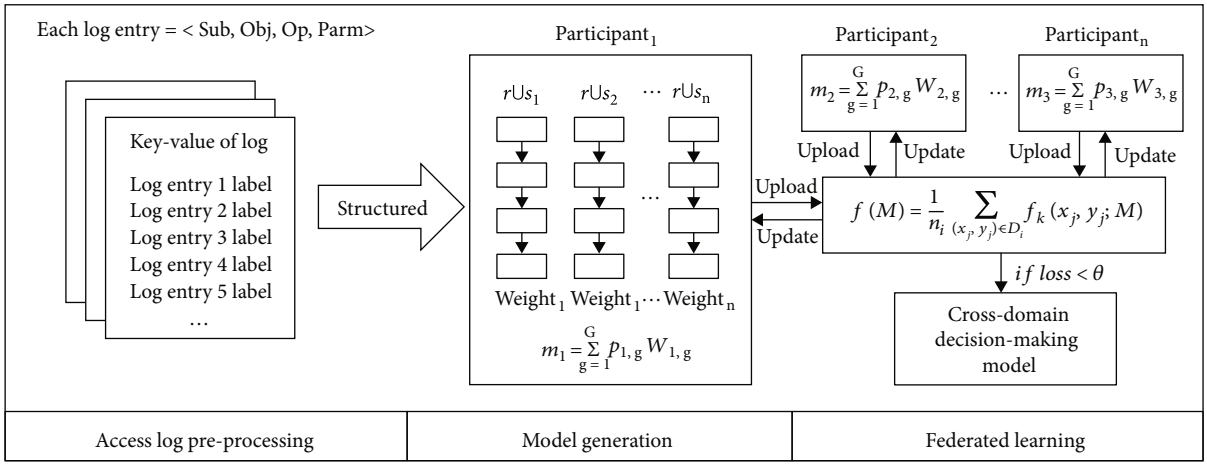

Figure 1: The workflow of FCAD.

the decision-making model without obtaining participants logs. This eliminates the privacy risk caused by the shared access information.

3. System Design

The workflow of FCAD is shown in Figure 1. In access log preprocessing, the crossdomain access logs are structured to get the access information in a key-value format. The information will be used as the training set of FCAD. Each participant generates its own model and shares the gradients of the model weights. After model generation, the central server collects the gradients from participants and updates the global model. The operation will be repeated until the loss function of the global model is satisfactory.

The system design of FCAD will be introduced in this section, including access log preprocessing, model generation, and federated learning.

3.1. Access Log Preprocessing. The crossdomain access logs are generated by different devices and systems, which causes their different formats. We propose an access log preprocessing method to regular the access logs. These logs are mainly generated with the information in crossdomain access requests, which can be defined as a four-tuple <*Subject, Object, Operation, Parameters*>. *Subject* represents the initiator of the access request. *Object* represents the resource or service being accessed. *Operation* represents the operation such as reading and writing. *Parameters* represents the additional information of the access request, such as the time when the smart door lock is requested to be opened, or the temperature that you want to set when the smart air conditioner is turned on. Logs are generally recorded by natural language in systems. The method of converting natural language logs into key value has been introduced in many works [19]. In FCAD, the crossdomain access logs used for training have been transfer to key-value format by default. A log format example is shown in Figure 2.

Subject and *Object* represent the identifications of the requests source and destination. A comprehensive and accurate identity definition enhances the accuracy of the learning model, since the probability of being attacked and the priority of each participant are different. *Operation* and *Parameters* define an operation together. The security levels of operations are diverse, and the environment context information and command parameters existing in the *Parameters* also affect the decision-making results. Extracting these effective information is necessary to improving the accuracy of the decision-making model.

Most of these valid information are saved by natural language. The extracted information can be expressed as [*Character Types Data, Numerical Data, Label*]. The information in *Character Types Data* is discrete, which cannot be directly used in training. Text vectorization methods are needed in FCAD to transform information into numerical data. On the other hand, the access logs provided by each participant often have different labels. For example, if the participant provides an anomaly detection dataset generated by itself, the data labels may be *DosAttack*, *ForgeryAttack*, etc. Encoding these labels into numerical values directly will complicate model training and affects the effective of the learning algorithm. To address the problem, we transform the label into <0, 1>. 0 and 1 mean that the access request is allowed or denied, so as to simplify the model training.

3.2. Model Generation

3.2.1. Model Training. We design the FCAD model training process to get a high-accuracy access decision-making model. Based on the features of training dataset, we design a binary classifier based on supervised learning. There are many machine learning models that can achieve good binary classification results, such as random forest and gradient boosting machine [20]. The sequential model is used in FCAD to achieve a model aggregation. Multiple fully connected network structures are stack to complete the classification. The model structure of FCAD is shown in Figure 3. Four fully connected layers are used in FCAD. The activation function of the first three layers is *Relu*, which is the widely used rectified linear units [21]. The *Sigmoid* function is added in the last layer, which decides whether the access request is allowed according to the prediction result. Binary

SourceID: Heatingcontrol4
SourceAddress: /agent25/heatingcontrol4
SourceType: /thermostat
SourceLocation: Room_6
DestinationServiceAddress: /agent20/tempin20
DestinationServiceType: /sensorService
DestinationLocation: Room_1
AccessedNodeAddress: /agent20/tempin20
AccessedNodeType: /sensorService
Operation: Read
Parameters: 20
Normality: Normal

Figure 2: Example of the log format on the used DS2OS dataset.

crossentropy is used as the loss function [22]. The global average pooling [23] can be used to replace the fully connected layer, to reduce the number of parameters and the communication overhead in federated learning.

3.2.2. Data Enhancement. In access logs of crossdomain collaboration, the number of allowed requests is much more than denied requests, which will lead to imbalance of the dataset. For example, on the DS2OS dataset, the average malicious access rate of each participant is only 2%. This makes the model trained by participants overfitting easily. To address the problem, we resample the dataset to improve the effectiveness. The rare class samples and part of the abundant class samples on the original dataset are combined as the resampled dataset, and the preduction accuracy is used as the weight for model averaging. The model averaging process can be expressed as follows:

$$\begin{aligned} W_{kg} &= M_k\left(r_k \cup s_{kg}\right), \\ m_k &= \sum_{g=1}^{G} p_{kg} W_{kg}, \end{aligned} \tag{1}$$

where W_{kg} is the weight matrix of the g -th model trained by the k -th participant, M_k is the model trained by the k -th participant, r_k is the k -th rare class samples, s_{kg} is the k -th abundant class samples selected for the g -th round, m_k is the combined weight, and p_{kg} is the prediction accuracy of M_{kg}. In FCAD, for a dataset with 2% of malicious access, we randomly generate 50 resampled datasets. The 50 models obtained will be aggregated with their prediction accuracy weights to eliminate the imbalance of the original dataset.

3.3. Federated Learning. Although an access decision-making model for a single participant can be obtained, it brings privacy risk in crossdomain access decision-making. We use federated learning solve this problem.

3.3.1. Workflow. The workflow of federated learning in FCAD is as following. First, the central server counts the templates of the local datasets of each participant and performs parameter division and data alignment according to the format of <*Subject, Object, Operation, Parameters*>. The learning model will be selected according to the result. Each participant performs data preprocessing on its local dataset and trains model. The data enhancement is independently implemented by participants. Then, the participant encrypts the model weight gradients and transmits it to the central server. Like an adaptive process [24], the central server obtains the updated model parameters by using the gradient aggregation scheme, broadcasts the updated model gradient to the newly selected participant, and iterates above operations until the loss function of the model is satisfactory.

3.3.2. Gradient Aggregation. The weights of local models are determined by the participant's data size and the prediction accuracy. We use the impact factor δ to measure the contribution of each participant's model, which can be expressed as

$$\begin{aligned} \delta_k &= \frac{(n_k/n)P_k}{\sum_{k=1}^{K}(n_k/n)P_k} = \frac{n_k P_k}{\sum_{k=1}^{K} n_k P_k}, \\ \omega_{t+1} &= \sum_{k=1}^{K} \delta_k \omega_{t,k}, \end{aligned} \tag{2}$$

where δ_k is the model contribution of the k -th participant, and n_k is the size of the k -th local dataset. We obtain the model contribution by calculating the contribution of the dataset and the prediction accuracy P_k. The model contribution is used to measure the importance of the participant. It directly affects the update of the global model $\sum_{k=1}^{K}\delta_k\omega_{t,k} \longrightarrow \omega_{t+1}$. The global model continues to iterate until it satisfies the iteration termination condition, which can be expressed as

$$\text{loss}(o_t, L_t) = -\frac{1}{n}\sum_{i}\left(L_t[i] * \log\left(o_t[i]\right) + \left(1 - L_t[i]\right) * \log\left(1 - o_t[i]\right)\right) < \theta. \tag{3}$$

Among them, $\text{loss}(o_t, L_t)$ is the loss function of the global model. $L_t[i]$ is the prediction result of the i -th label of the t -th iteration. $o_t[i]$ is the i -th input of the t -th iteration, which is the predicted value in [0,1]. We set $\text{loss}(o_t, L_t) < \theta$ as the termination condition, which represents that the model is considered to have converged.

3.3.3. Privacy and Security. It is worth to discuss privacy and security problems in federated learning-based methods [25, 26]. To prevent problems such as model leakage, data poisoning, and sample exposure, methods such as homomorphic encryption can be used to ensure the safety of learning and data exchanging [27]. Although there are many methods to ensure the data security, side-channel attacks are difficult to defend [28]. For example, we can predict whether the house access logs have been changed by comparing the new model with the old one, so as to predict the activity time of the house owner. In FCAD, the aggregated model

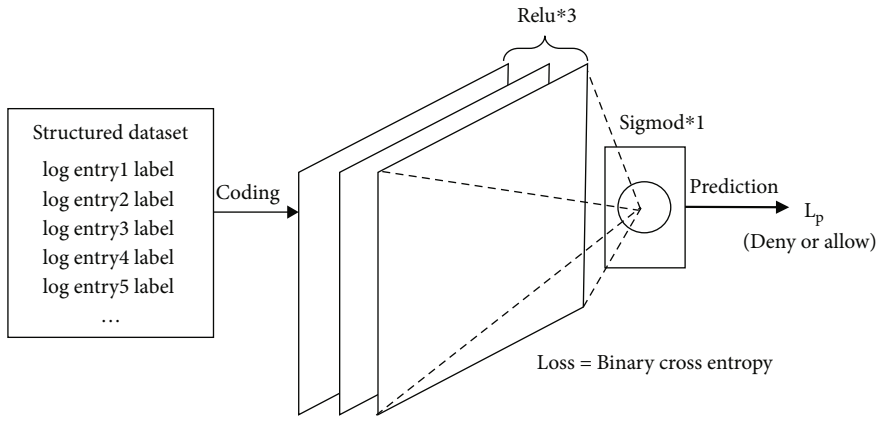

Figure 3: The model generation of FCAD.

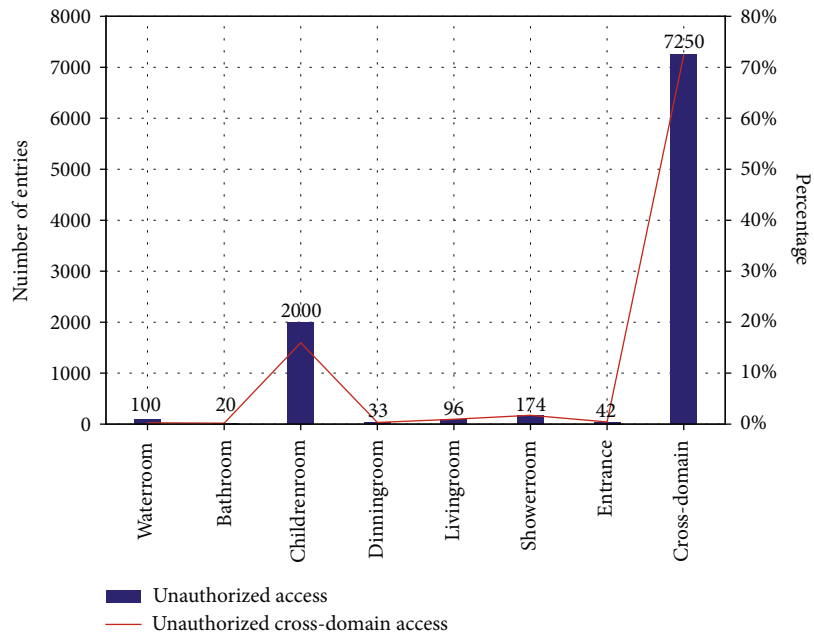

Figure 4: Part of abnormal crossdomain access percentage in the DS2OS dataset.

returned to participant changes almost every time, which is used as the old model in the next training round. Since the old model changes, the new participant model will also change, even if the same local access logs are used for training. This prevents the performing of side-channel attacks on FCAD.

4. Experimental Evaluation

We evaluate the prediction accuracy of FCAD in a public dataset to verify the effectiveness.

4.1. Dataset. We perform our experiment with the DS2OS traffic traces dataset [29]. The dataset contains access logs obtained in the IoT environment DS2OS. The crossdomain information are generated in application layers from four different simulated IoT sites. They provide different services, such as light controller, thermometer, movement sensors, washing machines, batteries, thermostats, smart doors, and smart phones. All the devices are implemented in 21 different locations. Each location is regarded as an independent domain. We count the percentage of crossdomain abnormal accesses on the DS2OS dataset, and the results are shown in Figure 4. Among the multiple domains existing in a smart home IoT environment, the percentage of crossdomain abnormal access is 72.4%. In the meantime, more than 7,000 abnormal crossdomain accesses have been made among the 350,000 access entries. The result shows that the DS2OS dataset is close to the real IoT crossdomain collaboration environment.

4.2. Implementation. After we extract the log key of DS2OS, the access log entries can be expressed as

<*Subject*>: {*SourceAddress*, *SourceType*, *SourceLocation*}.

<*Object*>: {*SestinationServiceAddress*, *DestinationServiceType*, *DestinationLocation*}.

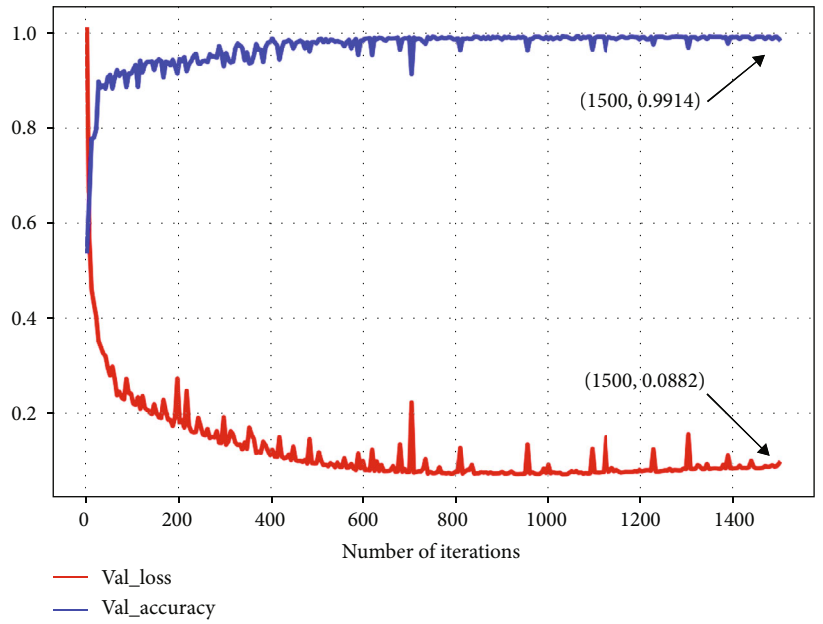

Figure 5: Model parameters of FCAD with the complete DS2OS dataset as the training set.

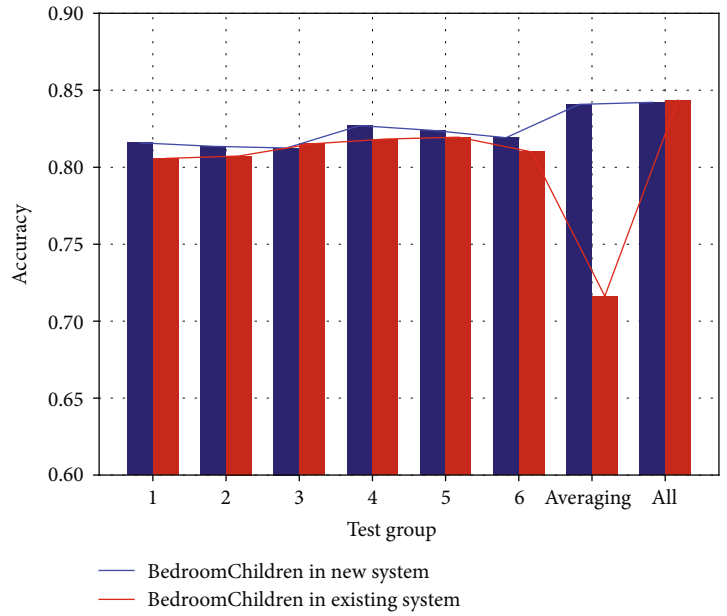

Figure 6: Different group accuracies of BedroomChildren local dataset.

Table 1: All the chosen proportions in new system and existing system.

Locations	Existing system	New system
Waterroom	3.155%	3.614%
Room_9	3.015%	7.517%
Bathroom	20.495%	0.723%
BedroomChildren	21.703%	72.280%
Entrance	9.334%	1.518%
Showerroom	19.856%	6.288%
Total	77.56%	85.65%

<*Operation*>: {*Operation*}.
<*Parameters*>: {*Value*, *Timestamp*}.

Since DS2OS has already identified the normality of access, we set the label of the normal access logs to 1, and the abnormal access logs to 0. We assume that all the normal access should be allowed, and all the abnormal access should be denied. The different locations are considered to be different domains and finally get 21 domains. We use two methods to obtain the local dataset of each domain. (i) Count entries with the same {*DestinationLocation*} as the local dataset and (ii) count the access logs within each domain as the local dataset (when

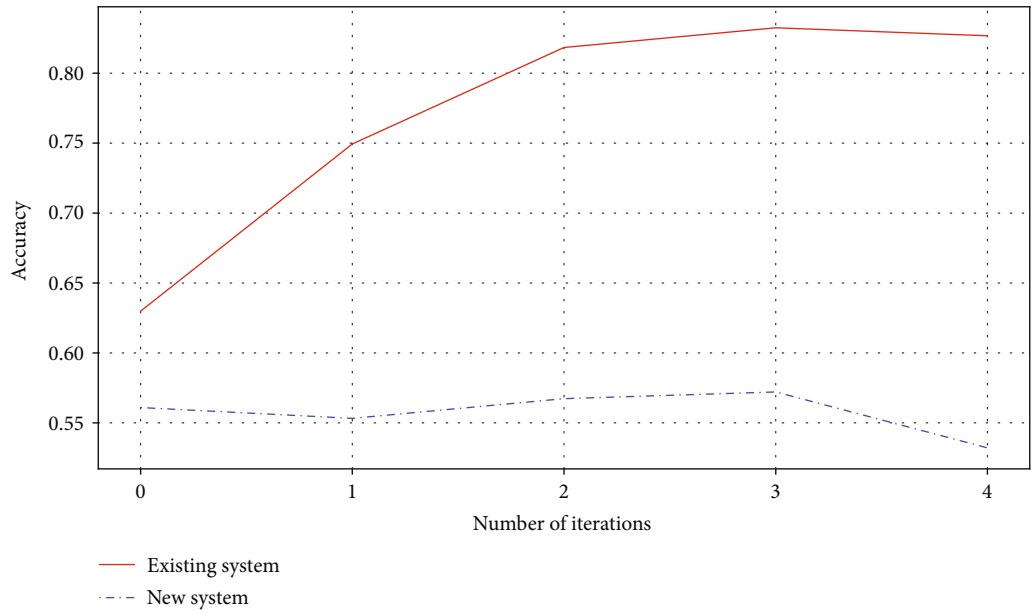

Figure 7: Effectiveness of gradient aggregation in FCAD.

{*SourceLocation*} = {*DestinationLocation*}). The two methods are designed to simulate two actual scenarios: the existing crossdomain collaboration system (which has crossdomain access logs) and the new crossdomain collaboration system (which has no crossdomain access logs). The first method is suitable for long-running crossdomain systems, since there are already a sufficient number of crossdomain access logs as the training set. The second method is suitable for newly constructed crossdomain systems, because there is no crossdomain access request as the training set. We use 30% of entries on the local dataset as the testing set. The dataset preprocessing algorithm is written in Python 3.5. The model training is completed by a laptop computer with a 1.60 Hz CPU (Intel i5-8250U) and 8GB RAM.

4.3. Experiments and Evaluation

4.3.1. Baseline. We use all the entries in our dataset to obtain the model training effect as our baseline. The result is shown in Figure 5. We find that the model converged when it iterates 1500 times, and it obtains a prediction accuracy of 99.14%. It shows the effectiveness of the neural network used by FCAD in predicting crossdomain access decisions.

4.3.2. Model Averaging Effectiveness. We choose *Bedroom-Children* as an example to evaluate the model averaging effectiveness. The obtained local dataset in existing system has 12768 entries, including 10594 positive samples and 2174 negative samples. The total proportion of negative examples is 17.02%. There are a total of 12,524 entries in the new system dataset, which has 10,524 positive samples and 2000 negative samples. The total proportion of negative examples is 15.96%. We generate 6 resampled datasets and use the same neural network for training. The result is shown in Figure 6. The numbers 1 ~ 6 represent the 6 resampled datasets, *Averaging* is the accuracy of the model, and *All* is the accuracy of the model trained on the original dataset without resampling.

We find that the model trained by single participant is far less effective than the model trained on the entire dataset. Although we have selected a participant with an even distribution, the model is still overfitting. The prediction accuracy is about 80%. Although the prediction accuracy of *All* is the highest, it is close to the original sample distribution. It shows that the model overfitting of *All* may be serious. For the existing system, the prediction accuracy is higher than that of single resampling data, because crossdomain access logs exist on the local dataset. It shows that the data enhancement method is effective to deal with uneven distribution. For the new system, it is difficult to get a good result based on intra-domain access logs; so, the prediction accuracy of the averaged model is greatly reduced.

4.3.3. Gradient Aggregation Effectiveness. We choose 6 participants with more negative samples for gradient aggregation. The result is shown in Table 1. For the existing system, the negative samples owned by these 6 participants accounted for 77.56% of all the negative samples. For the new system, it accounted for 85.65%. It is due to a more balanced occurrence of crossdomain abnormal access. We use the proportion of negative samples and the accuracy as the weight of the gradient aggregation.

The prediction accuracy of the access decision-making model after gradient aggregation is evaluated. The result is shown in Figure 7. For the existing system, the accuracy of the model will increase with the number of iterations, which can reach 83.6%. For the new system, the prediction accuracy is about 55%. The reason is that due to the lack of cross-domain access samples in the training dataset, the aggregated model is difficult to predict the decision-making results and almost loses its effect. It shows that FDAC is

more suitable for the existing crossdomain collaboration system.

5. Conclusion

In this paper, we propose a federated learning-based crossdomain access decision-making method FCAD. The designed log preprocessing method structures the crossdomain access logs to obtain the training dataset. Data enhancement method is used to address the distribution heterogeneity of the dataset which can be directly used in learning algorithms. Federated learning is used to prevent access logs sharing in crossdomain access decision model establishing. FCAD can obtain information highly relevant to the access decision from access logs and has a prediction accuracy of 83.6% in the existing crossdomain access system. The results show that making an access decision based on the access information obtained by the application layer is a feasible method in crossdomain collaboration. Since the decision made by the learning model can hardly be explained, and it is difficult to deal with updates of the access control policies, FCAD is more likely to be a supplement when access control policies are lacking.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by National Key R&D Program of China (No. 2018YFB2100400), National Science Foundation of China (Nos. 61872100, 62002077), Industrial Internet Innovation and Development Project of China (2019), State Grid Corporation of China Co., Ltd., Technology Project (No. 5700-202019187A-0-0-00), Guangxi Key Laboratory of Cryptography and Information Security (No. GXIS202119), Guangdong Basic and Applied Basic Research Foundation (No. 2020A1515110385), Zhejiang Lab (No. 2020NF0AB01), and Guangzhou Science and Technology Plan Project (No. 202102010440).

References

[1] C. Patsioura, *Blockchain and Distributed Ledger Technologies: what's the Value for IoT*, Technical Report. GSMA Intelligence, 2018.

[2] "IFTTT helps every thing work better together," May 2021, https://ifttt.com/.

[3] T. H. Payne, D. E. Detmer, J. C. Wyatt, and I. E. Buchan, "National-scale clinical information exchange in the United Kingdom: lessons for the United States," *Journal of the American Medical Informatics Association*, vol. 18, no. 1, pp. 91–98, 2011.

[4] "One simple home system. A world of possibilities. | SmartThings," May 2021, https://www.smartthings.com/.

[5] "Google Developers," May 2021, https://developers.google.com/.

[6] S. Ravidas, A. Lekidis, F. Paci, and N. Zannone, "Access control in internet-of-things: a survey," *Journal of Network and Computer Applications*, vol. 144, pp. 79–101, 2019.

[7] C. Xiang, Y. Wu, B. Shen et al., "Towards continuous access control validation and forensics," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 113–129, London, United Kingdom, 2019.

[8] P. Iyer and A. Masoumzadeh, "Mining positive and negative attribute-based access control policy rules," in *Proceedings of the 23nd ACM on Symposium on Access Control Models and Technologies*, pp. 161–172, Indianapolis, Indiana, USA, 2018.

[9] B. Yuan, Y. Jia, L. Xing, D. Zhao, X. Wang, and Y. Zhang, "Shattered chain of trust: understanding security risks in cross-cloud IoT access delegation," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pp. 1183–1200, Washington, D. C., 2020.

[10] Q. Li, B. Xia, H. Huang, Y. Zhang, and T. Zhang, "TRAC: traceable and revocable access control scheme for mHealth in 5G-enabled IIoT," *IEEE Transactions on Industrial Informatics*, p. 1, 2021.

[11] G. Neumann and M. Strembeck, "A scenario-driven role engineering process for functional RBAC roles," in *Proceedings of the seventh ACM symposium on Access control models and technologies*, pp. 33–42, Monterey, California, USA, 2002.

[12] Z. Xu and S. D. Stoller, "Mining attribute-based access control policies from logs," in *Data and Applications Security and Privacy XXVIII: IFIP Annual Conference on Data and Applications Security and Privacy*, Lecture Notes in Computer Science, pp. 276–291, Springer Link, 2014.

[13] C. Cotrini, L. Corinzia, T. Weghorn, and D. Basin, "The next 700 policy miners: a universal method for building policy miners," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 95–112, London, United Kingdom, 2019.

[14] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A survey of machine and deep learning methods for internet of things (IoT) security," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1646–1685, 2020.

[15] M. Narouei, H. Khanpour, and H. Takabi, "Identification of access control policy sentences from natural language policy documents," in *Data and Applications Security and Privacy XXXI: IFIP Annual Conference on Data and Applications Security and Privacy*, Lecture Notes in Computer Science, pp. 82–100, Springer Link, 2017.

[16] D. Mocanu, F. Turkmen, and A. Liotta, *Towards ABAC Policy Mining from Logs with Deep Learning*, University of Groningen, 2015.

[17] L. Karimi and J. Joshi, "An unsupervised learning based approach for mining attribute based access control policies," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 1427–1436, Seattle, WA, USA, 2018.

[18] A. A. Jabal, E. Bertino, J. Lobo et al., "Polisma-a framework for learning attribute-based access control policies," in *Computer Security – ESORICS 2020: European Symposium on Research in Computer Security*, Lecture Notes in Computer Science, pp. 523–544, Springer Link, 2020.

[19] M. Du, F. Li, G. Zheng, and V. Srikumar, "Deeplog: anomaly detection and diagnosis from system logs through deep learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1285–1298, Dallas, Texas, USA, 2017.

[20] J. H. Min and C. Jeong, "A binary classification method for bankruptcy prediction," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5256–5263, 2009.

[21] A. F. Agarap, "Deep learning using rectified linear units (relu)," 2018, https://arxiv.org/abs/1803.08375.

[22] U. Ruby and V. Yendapalli, "Binary cross entropy with deep learning technique for image classification," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 4, pp. 5393–5397, 2020.

[23] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, https://arxiv.org/abs/1312.4400.

[24] P. Zhu, Q. Zhi, Y. Guo, and Z. Wang, "Analysis of epidemic spreading process in adaptive networks," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 66, no. 7, pp. 1252–1256, 2018.

[25] J. Xiong, R. Bi, Y. Tian, X. Liu, and D. Wu, "Towards lightweight, privacy-preserving cooperative object classification for connected autonomous vehicles," *IEEE Internet of Things Journal*, p. 1, 2021.

[26] J. Xiong, R. Ma, L. Chen et al., "A personalized privacy protection framework for mobile crowdsensing in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4231–4241, 2019.

[27] C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, "Batchcrypt: efficient homomorphic encryption for cross-silo federated learning," in *2020 {USENIX} Annual Technical Conference ({USENIX}{ATC} 20)*, pp. 493–506, Boston, 2020.

[28] F. Mo and H. Haddadi, *Efficient and Private Federated Learning Using Tee*, Proc. EuroSys Conf., Dresden, Germany, 2019.

[29] "DS2OS traffic traces," 2021, https://kaggle.com/francoisxa/ds2ostraffictraces.