2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

A groundbreaking and innovative data privacy protection framework

Binghui Liu[1*]

Xinjiang Teacher's College

17796721009@163.com

Abstract. In scenarios like smart healthcare, smart communities, and smart buildings, data collected by Internet of Things (IoT) devices often pertains to user privacy. However, due to the limited computing power and storage capacity of IoT devices, the data of Data Subject (DS) are generally stored in the cloud, causing DS to lose control over his data and increasing the risk of privacy leakage. Additionally, resource-constrained IoT devices often face affordability issues regarding encryption costs. In this paper, we propose PPFID, an efficient privacy preserving framework with the DS's intentions. Specifically, PPFID enforces isolated computation and permission control via secure enclaves of Intel SGX on centrally aggregated data, and encrypts data to guarantee confidential access, computation, and delivery throughout the entire life of the data. To support fine grained access control with the wishes of DS as its core, we design the Privacy Metadata-Based Access Control (PMBAC) model, which consider the wishes of DS to make access control decisions for each piece of data. Compared to other schemes, PPFID provides more data processing methods and introduces access control schemes that are both strongly isolated and respect DS's rights. We successfully implemented PPFID on Intel SGX and the embedded device, and evaluated the its feasibility. Our evaluation shows PMBAC can process an access request in the enclave in just 140ms, meeting DS's real-time requirements. Al- though the computing time has increased compared to the non-protected environments, the prediction accuracy of VGG19 and CNN remains essentially the same. Experimental results demonstrate that PPFID is applicable in general IoT scenarios involving users' privacy data, and can ensure the confidentiality, integrity, and availability of data while respecting the wishes of DS.

Keywords: Internet of Things, IoT Privacy Preserving, Intel SGX

1 Introduction

The Internet of Things (IoT) are closely related to people's daily lives [1], and its integration with cloud computing is crucial for data management. However, given the distributed

features of IoT, the issue of personal privacy data leakage becomes more pronounced during the processes of transmission, storage, and usage [2]. For example, Avanti Markets [3] suffered a malware attack on its internal network, resulting in the theft of users' biometric information.

Although the regulations like GDPR [4] and the APRA [5] to protect privacy and grant Data Subject (DS) the ultimate control of privacy data (PD), most users lack awareness of privacy and security. A study by the Pew Research Center [6] found that many Americans are overly optimistic about the ways their data is used. Only 26% of Americans reject to share their health information with doctors. To meet the demands for security and privacy, IoT device manufactures, cloud providers, and researchers are working to design security systems and to seek effective ways to protect data privacy [7]. Such as employing third-party managers or encryption [8], distributed collaborative training [9], Federated Learning [10] and access control [11]. There is an emerging trend to- wards leveraging Trusted Execution Environments (TEE), or isolated enclaves, to secure machine learning training pipelines. For example, Ohrimenko et al. [12] proposed using Intel Software Guard Extensions (SGX) to enable multi-party training for different Machine Learning (ML) methods. More recently, Chiron [13] and Myelin [14] integrated SGX to support private Deep Learning (DL) training services. In recent years, access control has emerged as a crucial security mechanism. Existing access control schemes of the IoT do not pay attention to the security problems, and once the access control of the IoT is broken, it will cause serious consequences such as privacy data leakage and authority abuse. Liu et al. [15] achieve Attribute-based Access Control, which allows DS to have different access authorities to attribute values. Ciphertext policy attributed based encryption (CPABE) , which allows users to formulate access control policies, is flexible and becomes the mainstream technology for user data security in a cloud storage environment. Existing schemes have the issue of limited applicability. Most of solutions are only applicable to specific application scenarios, such as smart grid, smart healthcare, Internet of vehicles, etc., or only apply to a certain stage of the data life cycle, such as data collection and sharing of private data with cloud services. Additionally, they often fail to consider the personal preferences of DS, such as whether to agree to the addition, deletion modification and inspection of data. There is an emerging trend towards leveraging Trusted Execution Environments (TEE), or isolated enclaves, to secure machine learning training pipelines. For example, Ohrimenko et al. [12] proposed using Intel Software Guard Extensions (SGX) to enable multi-party training for different Machine Learning (ML) methods. More recently, Chiron [13] and Myelin[14] integrated SGX to support private Deep Learning (DL) training services. In recent years, access control has emerged as a crucial security mechanism. Existing access control schemes of the IoT do not pay attention to the security problems, and once the access control of the IoT is broken, it will cause serious consequences such as privacy data leakage and authority abuse. Liu et al. [15] achieve Attribute-based Access Control, which allows DS to have different access authorities to attribute values. Ciphertext policy attributed based encryption (CPABE) [26], which allows users to formulate access

control policies, is flexible and becomes the mainstream technology for user data security in a cloud storage environment. Existing schemes have the issue of limited applicability. Most of solutions are only applicable to specific application scenarios, such as smart grid, smart healthcare, Internet of vehicles, etc., or only apply to a certain stage of the data life cycle, such as data collection and sharing of private data with cloud services. Additionally, they often fail to consider the personal preferences of DS, such as whether to agree to the addition, deletion modification and inspection of data.

Finally, they often focus solely on protecting DS's data while neglecting the protection of data related to access control models. IoT devices are generally not strongly bound to individual users, and within IoT scenarios, there is a need to process both personal data and aggregate public data. This necessitates a data protection scheme that facilitates data processing while safeguarding individual privacy. To tackle these issues, this paper focuses on common smart IoT scenarios and proposes a privacy-preserving framework, PPFID, which takes the wishes of each DS as the core. We encrypt the PD on IoT devices to ensure that PD is securely uploaded to the cloud. Then, we design a Privacy Metadata-Based Access Control (PMBAC) model based on the wishes of DS. PMBAC is executed within the securely isolated Enclave, which not only safeguards the access policies but also protects PD from over-calculation and unauthorized access. To further process the privacy protection problem of data in use, we introduce confidential computing and experimentally verify the feasibility of various data processing methods. We establish a secure chain for the entire lifecycle of PD from generation to destruction While respecting DS's wishes. Our contributions are summarized as follows.

–PMBAC: PMBAC achieves fine-grained access control based on the wishes of DS and comprehensive policy preserving, ensuring the non-disclosure of any sensitive information in the PD and access policy.

–Confidential Learning: To protect data confidentiality during use, we achieve a TEE-based data processing system in the Enclave, supporting statistical computing, machine learning (ML) and deep learning (DL).

–Verify: We implement PPFID on Intel SGX and IoT de vice, and systematically evaluate the additional overhead associated with each module, including data encryption on the IoT device, confidential computing and PMBAC.

The rest of this paper is as follows: Section 2 analyzes and compares existing privacy protection schemes in IoT scenarios. Section 3 defines design objectives and introduces the PPFID. Section 4 details the implementation of PPFID. Section 5 presents experiments and performance evaluations to verify the feasibility of PPFID. Section 6 summarizes the research work presented in this paper.

2 Related Work

This section introduces some existing privacy-preserving schemes and technologies in IoT scenarios and compares them with PPFID, with the detailed comparison shown in Table I. Lightweight encryption algorithms [16] are designed to encrypt PD transmitted in the IoT with low performance overhead. These algorithms ensure that even if the data is intercepted or analyzed through packet capture, it is impossible to extract plaintext data or infer related information from the captured packets. Implementing fine grained access control mechanisms [17] between receiving and transmitting devices in the IoT can effectively prevent the overuse and unauthorized access of IoT data. Confidential computing [18] often leverages special hardware designs to implement a secure TEE, where data and its processing procedures are deployed to protect data privacy and security. Given the rather singular nature of privacy-preserving technologies, many scholars have begun to combine various privacy protection techniques to construct effective privacy protection schemes. This approach aims to achieve better privacy protection outcomes, thereby reducing the risk of data breaches. Li et al. [19] developed a lightweight privacy protection scheme based on homomorphic encryption, which encrypts data collected by sensors and uploads it to a third-party cloud server. When access to the data is required, the cloud server decrypts the data and returns the results. Zhao [20] utilized elliptic curve cryptography and hash functions to implement a lightweight data-sharing scheme, and introduced access control mechanisms to protect data. Although both schemes can prevent privacy leaks in IoT, they are not capable of conducting complex computations. Zhu et al. [21] proposed an IoT-accessible cloud-edge collaborative solution, employing chaotic mapping algorithms for efficient authentication, ensuring user anonymity and no traceability. The scheme can achieve efficient authentication and key negotiation in cloud-edge collaborative network architectures. Although the scheme can ensure data security, it is unable to provide precise authorization for each data operation. Zhang [22] proposed an IoT data privacy supervision compliance scheme, PACTA, which stores encrypted data on the blockchain. When the Data User (DU) accesses data, PACTA first checks whether the DS is online. If the DS is online, PACTA will ask for the DS's opinion and handle the access request accordingly. If the DS is offline, the system's policy will be followed for authorization. However, this scheme cannot guarantee that the DS always has the right to be informed about their data. Valadares [23] proposed a general IoT architecture based on TEE, employing authentication, authorization, and encryption mechanisms to ensure data confidentiality and integrity. However, all data uses the same verification and authorization methods, which does not allow for more finegrained authorization, and also fails to guarantee the DS's right to be informed about their data. Xie [24] deployed a privacy protection framework on IoT devices, which introduces identity authentication and hash technology to ensure the legitimacy of terminal devices during the data collection phase and the integrity of the data. The framework further designs a local differential privacy algorithm based on the proportion difference of feature information to protect privacy data transmitted between

edge devices and the cloud, ensuring data security within edge devices. Despite these measures, it does not protect data on the cloud.

3 SYSTEM OVERVIEW

3.1 Design Goals

We propose PPFID, a privacy-preserving framework for IoT data, to ensure security during transmission, storage and usage. PPFID should meet the following requirements:

–Implement fine-grained informed access control for each piece of personal data within a public aggregation database. This ensures precise authorization between access requests and individual PD, reducing the risk of data leakage or misuse due to over-authorization.

–While safeguarding the privacy and confidentiality of IoT data, the solution.

should efficiently support various data processing models in the cloud. This ensures the confidentiality of data during usage and enables rapid processing of DU's requests.

–Achieve secure transmission of data and secure delivery of data processing results.

3.2 Overview of PPFID

PPFID utilizes encryption middleware to encrypt PD and PM, ensuring the security of data in transit. After the cloud platform receives the data, it stores the data directly in ciphertext form on the cloud. Subsequent data processing will be carried out within the Enclave, which is inaccessible to others, ensuring that no one else can see the data within. These can guarantee the security of data during storage and usage on the cloud platform. Additionally, PMBAC, based on the PM of DS, enforces fine-grained access control over PD, with the DS's wishes playing a pivotal role in determining the permissions for data access. The system architecture is depicted in Figure 1.

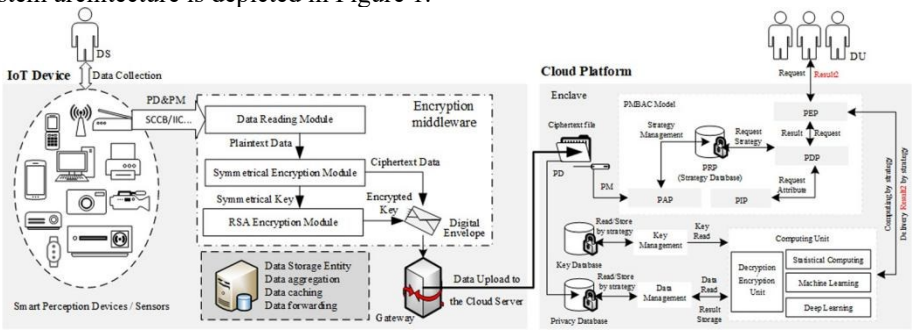


Fig. 1. PPFID Structure

Data Subject: The owner of the data, who wishes to securely store PD collected by IoT devices in the cloud.

Data User: The entity that utilizes the data, capable of sending access re- quests to the cloud. The cloud devices then return results based on access control policies.

IoT Device: It can sense the surrounding environment and serves as the source of data generation. It is responsible for encrypting the data and securely transmitting it in ciphertext form to the cloud for storage and usage. Encryption Middleware: A trusted entity on the IoT de vice encrypts PD and PM, ensuring that PD is transmitted in encrypted form. Gateway: A crucial node in the IoT system, connecting IoT devices and cloud servers. It is responsible for receiving, aggregating, caching, and forwarding the encrypted data. Cloud Platform: Achieves fine-grained secure access, confidential computing, and confidential delivery of PD through the PMBAC and computing unit.

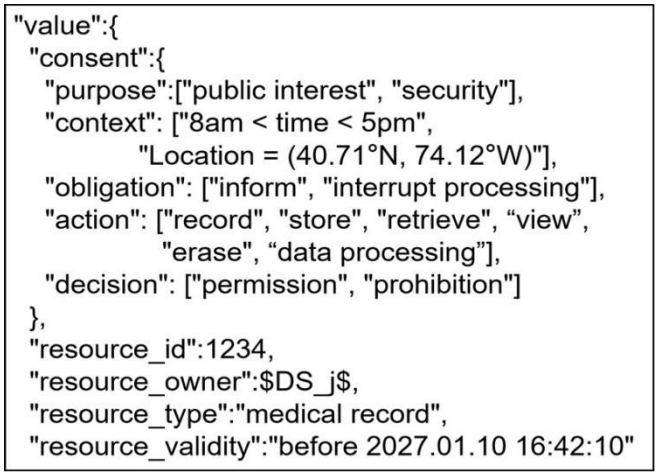


Fig. 2. PPFID Structure

4 CONSTRUCTION OF OUR WORK

4.1 IoT Data Collection and Informed Consent

Given the presence of a large amount of sensitive information in IoT scenarios, the collection, transmission, storage, and use of data must be handled with caution to ensure the security of PD. To ensure that the storage and use of data do not deviate from the DS's intentions, a standardized IC is required before uploading data. Once DS fills out and submits the IC, the system immediately generates corresponding PM based on the IC. PM is the primary basis for setting data access permissions. When DU access the data, the

system executes subsequent operations based on the authorization results of the PMBAC model.

4.2 IoT Data Encryption, Transmission, and Reception

After IoT devices collect PDj and ICj, they immediately generate PMj based on the content of ICj and encrypt PDj and PMj through IoT encryption middleware. Subsequently, PDEj, PMEj, and KCipher are combined into a digital envelope EDigital, which is uploaded to the cloud platform through the gateway to ensure the confidentiality of data in transit. Even if the EDigital is intercepted during transmission, the interceptor cannot obtain the Prij and KAES required for decryption, and may not even be able to distinguish which part is PDj. The cloud receives the EDigital and securely receives it into the exclusive memory of the cloud device in ciphertext form through the Intel SGX scheduler, where it is stored until data destruction, based on current business needs and access control policies. By adopting encryption middleware, efficient, secure, and reliable encryption of IoT data can be achieved, ensuring the confidentiality of data in transit. In practical applications, encryption middleware should have good compatibility, flexibility in different IoT scenarios, and the ability to transparently encrypt and transmit IoT privacy data. The cloud communication center needs to have strong data caching and transmission capabilities to effectively handle a large volume of encrypted messages and files, thereby ensuring the stability and security of the system.

4.3 PMBAC and Computing Unit

The PMBAC model and the computing unit are constructed on the cloud plat- form with Intel SGX. Intel SGX can directly generate a secure memory space called Enclave through the CPU. Code and data within the Enclave are stored in an encrypted form, and programs can execute securely within it. The Enclave provides enhanced security protection at the hardware level that can effectively block attacks even if the underlying software or systems are compromised. When running within an Enclave, its privileges are higher than any privileged software or system software, preventing malicious programs from tampering with or eaves- dropping on the Enclave, thus protecting the confidentiality and integrity of data and code stored or executed within it. Within the Enclave, after decrypting the EDigital, PDj and PMj are obtained. PMj is a set of authorizations from DSj for PDj and serves as the primary basis for setting access control policies. PDj is processed in the data processing module based on the authorization results from PMBAC, and the results of data processing are delivered to DU.

PMBAC: The PMBAC shares similarities with Attribute Based Access Control (ABAC) [34]. The key differentiator of the PMBAC model is the delegation of partial attribute management to the DS, rather than relying solely on administrators. This shift in control aims to enhance user autonomy, flexibility, and the overall security and usability of the

access control system. PMBAC is responsible for generating access control policies based on PMj and processing DU's access requests to prevent privacy leaks due to excessive computation and unauthorized data access.

When a DS uploads PD, DS is required to complete the IC. Following the submission of the IC, the system immediately generates PM based the contents of the IC. Both PD and PM are encrypted and subsequently uploaded to the cloud. The PD is securely stored within the TEE of the cloud, while the PM contributes to the formulation of the strategy database. The system to manage the storage and processing of data according to access control policies. If DS wants to alter the access permissions of the data, they simply need to find the corresponding IC form, make changes, and resubmit it for updates.

When a DU requests accesses to data, the PDP gives an authorization decision based on the messages obtained from queries to the PIP and the PRP. Subsequently, the PEP carries out the necessary data processing in accordance with the authorization decision. Finally, the results of data processing are encrypted, and the encrypted data is sent to DU.

5 EXPERIMENTS

To verify the feasibility of PPFID, we implement PPFID using Intel SGX and the embedded device, and evaluate PPFID based on real-world and publicly available datasets. Specifically, we evaluate and compare the computational cost in the context of both the application of the PPFID and its absence, and the accuracy of DL models in the Enclave.

5.1 Experimental Setup

Our experiments are performed on a Linux servers with Intel i7-10700 CPU running at 2.90 GHz with 16 threads on 8 cores and 7.5 GB memory, and a embedded device with ARM 32-bit CortexTM-M4 CPU running at 168 MHz. We use 1024-bit RSA for asymmetric encryption and signature, 128 bit AES for encrypting PD and PM on the embedded device, use the Pandas, Sklearn, Tensorflow, Keras to achieve many data processing methods, and use the Casbin to build PMBAC model. The experiment consisted of three parts. (A) We implement the processes of information collection, processing, and transmission on the embedded devices, documenting the time and power costs both with and without the introduction of encryption. (B) We achieve the statistical computing, machine learning, and deep learning both within the Enclave and on the Linux servers, and compare the computation cost of the same data processing methods and the accuracy of DL models in different environments. (C) In order to compare the computation cost of process a request in different environments, we generate a strategy database containing 1,000,000 policies and achieve the PMBAC model both within the Enclave and on the Linux servers.

Table 1. The Computational Cost of IoT Devices

	Time	Power
no-encryption	29 ms	0.700 W
encryption	32 ms	0.754 W

5.2 Datasets

The A dataset is collected in real time by embedded devices. The B dataset used in statistical calculation is derived from the collected data in A, following a process of organization and annotation (10 categories). The dataset in ML is a cleaned and consolidated Diabetes dataset [25] created from the dataset was released by the Centers for Disease Control and Prevention, including (a) data1: n = 253680, d = 21, hl = 3; (b) data2: n = 70692, d = 21, hl = 2; (c) data3: n = 253680, d = 21, hl = 2. The dataset in DL: (a) Breast ultrasound images dataset [26]: n = 1578, d = (128, 128, 3), hl = 3; (b) ChestX-ray images dataset [27]: n = 5240, d = (32, 32, 3), hl = 4; (c) skin cancer images dataset [27]: n = 3297, d = (128, 128, 3), hl = 2. Here, n represents the number of samples, d represents the number of input features, and hl represents the number of categories.

we generate a strategy database containing 1,000,000 policies based on the information contained in the PM.

5.3 Experimental Results

Impact of Encryption on the Performance of IoT Device: A power detector is connected in series with the embedded devices, and measurements are taken continuously for 2 hours, with sampling every 30 seconds. The average value of 240 sampling results is calculated to determine the average power. Introducing encryption results in an increase of 3ms in execution time and an increase of 0.054 W in power consumption. Considering the importance of data security, the increases in time and power are small, indicating that the impact of the encryption operation on the devices performance is minimal.

Statistical Computing: Ten datasets of different sizes but with the same data format and composition are loaded both inside and outside the Enclave, and the average value of each category is calculated. The time to load the data and the time consumed by the calculation are recorded. After testing each dataset 100 times, the results are averaged, and the experimental results are shown in Figure 3.

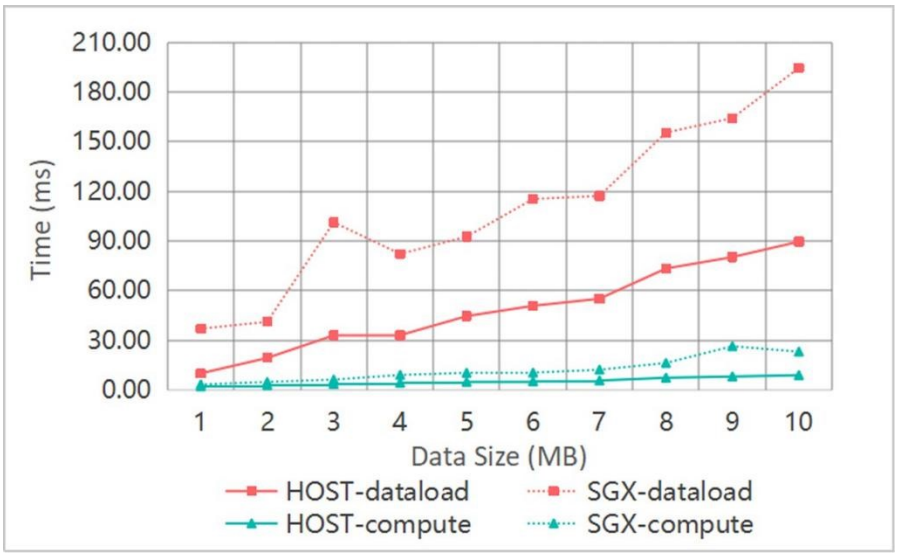
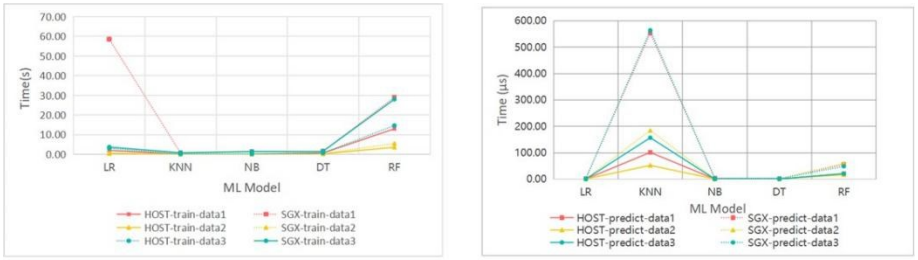


Fig. 3. Comparison of Enclave's internal and external Statistical Computing costs.

Machine Learning: The same models and datasets are used inside and outside the Enclave. Logistic Regression (LR), K-Nearest Neighbor (KNN), Naive Bayes (NB), Decision Trees (DT), and Random Forests (RF) are trained and used to predict three datasets of different sizes and categories. Each model is trained for 100 epochs, the average time overhead of one epoch is recorded in Figure 4. The model training time within the Enclave is kept within 60 s, and the model prediction time overhead is also at the microsecond level, meeting real-time requirements.

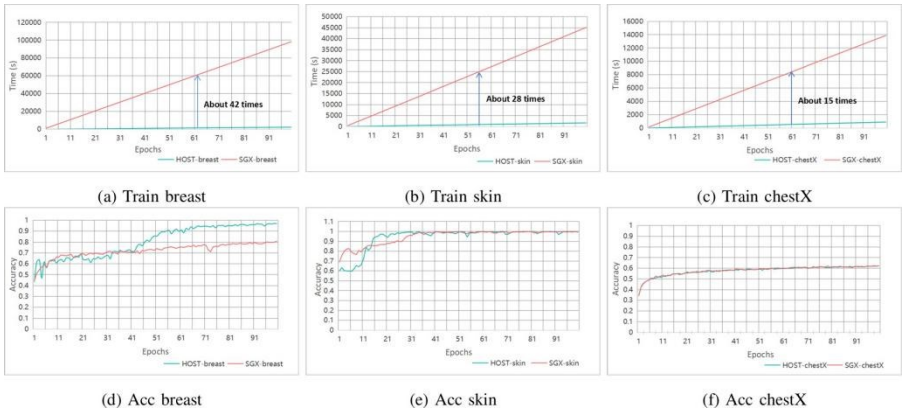


(a) Train (b) Predict

Fig. 4. The Training and Prediction Costs of ML

Deep Learning: Transformer, Visual Geometry Group 19 (VGG19), and Convolutional Neural Network (CNN) models, all built with the TensorFlow deep learning framework,

are used to train and predict breast ultrasound images, ChestX-ray images [37], and skin cancer images, respectively. To ensure the successful execution of the models within the enclave, we have appropriately reduced the model size and allocated sufficient resources. The datasets and models used inside and outside the enclave are consistent. Each model is trained for 100 epochs, and the time overhead and accuracy changes are shown in Figure 5. more complex the model, the greater the multiple of training duration increase.



(a) Train breast (b) Train skin (c) Train chestX

(d) Acc breast (e) Acc skin (f) Acc chestX

Fig. 5. The Training and Prediction Costs of ML

As for accuracy, due to the Enclaves limited memory of only 128 MB, if the model is too complex, gradient vanishing or exploding phenomena may occur during the training process. Therefore, models with higher complexity show a slight decrease in accuracy; models with lower complexity show almost no change in accuracy. Multiple model inference tasks are repeated, and the average time overhead is recorded as the duration required for one prediction. It is found that the prediction duration for each model does not exceed 500ms, meeting user requirements for real-time performance.

PMBAC: The PMBAC model proposed in this paper is implemented based on the Casbin 1 access control framework, and the model runs both inside and outside the Enclave to compare and analyze the impact of the Enclave. We recorded the total duration for PMBAC to process 1000 access requests, and the average duration is taken as the time to process one access request. The PMBAC model processes a request in no more than 112ms outside the Enclave and no more than 140ms inside the Enclave, with the time overhead increase not exceeding 25%. We have experimentally evaluated and compared the above scheme from aspects such as the computational cost of the encryption, the data processing, and the PMBAC processing one access request. The results show that after adopting this scheme, the time overhead increase in each part is not significant, and for neural networks with low complexity, there is almost no decrease in model prediction accuracy inside and outside the Enclave, proving the feasibility of the scheme.

6 CONCLUSIONS

First of all, an encryption middleware for IoT devices is designed, capable of securely transmitting IoT privacy data to the IoT cloud platform. Subsequently, PPFID innovatively integrates TEE, a hardware security technology, into the cloud platform to achieve secure isolation and protection of the data processing procedures, preventing attackers from snooping or stealing data. Following this, a PMBAC model is designed based on respecting user intentions, enabling fine grained authorization for various user operations such as viewing, deleting, modifying, and utilizing data. Finally, the performance overhead of the IoT encryption middleware and key technical modules of the secure isolation space is evaluated and validated. The results demonstrate that the PPFID can effectively adapt to various data analysis services and successfully employ trusted computing and access control technologies to protect user privacy data.

References

1. Kumar, R., Kumar, P., Tripathi, R., Gupta, G.P., Garg, S., Hassan, M.M.: Bdtwin: An integrated framework for enhancing security and privacy incybertwin-driven automotive industrial internet of things. IEEE Internet of Things Journal 9(18), 17110–17119 (2021)
2. Hameed, A., Alomary, A.: Security issues in iot: A survey. In: 2019 International conference on innovation and intelligence for informatics, computing, and technologies (3ICT). pp. 1–5. IEEE (2019)
3. Syed, A., Purushotham, K., Shidaganti, G.: Cloud storage security risks, practices and measures: A review. In: 2020 IEEE international conference for innovation in technology (INOCON). pp. 1–4. IEEE (2020)
4. Nagarajan, G., Kumar, K.S.: Security threats and challenges in public cloud storage. In: 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). pp. 97–100. IEEE (2021)
5. Pishva, D.: Internet of things: Security and privacy issues and possible solution. In: 2017 19th international conference on advanced communication technology (ICACT). pp. 797–808. IEEE (2017)
6. Voigt, P., Von dem Bussche, A.: The general data protection regulation (gdpr). A practical guide, 1st ed., Cham: Springer International Publishing 10(3152676), 10–5555 (2017)
7. Peltz-Steele, R.J.: The new American privacy. Geo. J. Int'l L. 44, 365 (2012)
8. Kim, C.H., Kim, T., Choi, H., Gu, Z., Lee, B., Zhang, X., Xu, D.: Securing real-time microcontroller systems through customized memory view switching. In: NDSS (2018)
9. Yang, L., Humayed, A., Li, F.: A multi-cloud based privacy-preserving data publishing scheme for the internet of things. In: Proceedings of the 32nd annual conference on computer security applications. pp. 30–39 (2016)
10. Guan, L., Xu, J., Wang, S., Xing, X., Lin, L., Huang, H., Liu, P., Lee, W.: From physical to cyber: Escalating protection for personalized auto insurance. In: Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD- ROM. pp. 42–55 (2016)

11. Mazon-Olivo, B., Pan, A.: Internet of things: state-of-the-art, computing paradigms and reference architectures. IEEE Latin America Transactions 20(1), 49–63 (2021)
12. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics. pp. 1273–1282. PMLR (2017)
13. Merlec, M.M., In, H.P.: Sc-caac: A smart contract-based context-aware access con- trol scheme for blockchain-enabled iot systems. IEEE Internet of Things Journal (2024)
14. Ohrimenko, O., Schuster, F., Fournet, C., Mehta, A., Nowozin, S., Vaswani, K., Costa, M.: Oblivious {Multi-Party} machine learning on trusted processors. In: 25th USENIX Security Symposium (USENIX Security 16). pp. 619–636 (2016)
15. Hunt, T., Song, C., Shokri, R., Shmatikov, V., Witchel, E.: Chiron: Privacy- preserving machine learning as a service. arXiv preprint arXiv:1803.05961 (2018)
16. Hynes, N., Cheng, R., Song, D.: Efficient deep learning on multi-source private data. arXiv preprint arXiv:1807.06689 (2018)
17. Liu, Y., Yu, J., Fan, J., Vijayakumar, P., Chang, V.: Achieving privacy-preserving dsse for intelligent iot healthcare system. IEEE Transactions on Industrial Informatics 18(3), 2010–2020 (2021)
18. Yu, Y., Guo, L., Liu, S., Zheng, J., Wang, H.: Privacy protection scheme based on cp-abe in crowdsourcing-iot for smart ocean. IEEE Internet of Things Journal 7(10), 10061–10071 (2020)
19. Dwivedi, A.D., Srivastava, G.: Security analysis of lightweight iot encryption algorithms: Simon and simeck. Internet of Things 21, 100677 (2023)
20. Wang, G., Li, C., Dai, B., Zhang, S.: Privacy-protection method for blockchain transactions based on lightweight homomorphic encryption. Information 15(8), 438 (2024)
21. Tiwari, D., Mondal, B., Singh, S.K., Koundal, D.: Lightweight encryption for privacy protection of data transmission in cyber physical systems. Cluster Computing 26(4), 2351–2365 (2023)
22. Agrawal, M., Zhou, J., Chang, D.: A survey on lightweight authenticated encryption and challenges for securing industrial iot. Security and privacy trends in the industrial internet of things pp. 71–94 (2019)
23. Ouaddah, A., Elkalam, A.A., Ouahman, A.A.: Towards a novel privacy-preserving access control model based on blockchain technology in iot. In: Europe and MENA cooperation advances in information and communication technologies. pp. 523–533. Springer (2017)
24. Russinovich, M.: Confidential computing: Elevating cloud security and privacy. Communications of the ACM 67(1), 52–53 (2023)
25. Lee, D., António, J., Khan, H.: Privacy-preserving decentralized ai with confidential computing. arXiv preprint arXiv:2410.13752 (2024)
26. Chang, V., Ganatra, M.A., Hall, K., Golightly, L., Xu, Q.A.: An assessment of ma- chine learning models and algorithms for early prediction and diagnosis of diabetes using health indicators. Healthcare Analytics 2, 100118 (2022)
27. Gheflati, B., Rivaz, H.: Vision transformers for classification of breast ultrasound images. In: 2022 44th annual international conference of the IEEE Engineering in Medicine & Biology Society (EMBC). pp. 480–483. IEEE (2022)