A Privacy and Efficiency-Oriented Data Sharing Mechanism for IoTs

Chao Wang, *Member, IEEE*, Shuo Wang, Xiaoman Cheng, Yunhua He, *Member, IEEE*, Ke Xiao, and Shujia Fan

Abstract—With the volume of data increasing in the Internet of Things, a new business mode, where data owners share their own data to others for rewards, has emerged. Therefore, how to motivate data owners to participate in the data trading process is the main challenge. So far, lots of works focus on the motivation mechanism designing and ensure a fair distribution of profits among data owners. However, some security and privacy issues are still not well solved and the data owners are still unwilling to participate in the process. Especially, when a data provider claims rewards with its real identity for the shared data, the linkage between its real identity and the shared data will expose the participator's private information included in the shared data, such as location information. To protect user's privacy in the scenario, a privacy and efficiency-oriented data sharing mechanism for IoTs is proposed in this paper. We first propose a blockchain-based data sharing framework in which the behavior of all participants will be supervised. Then, in order to hide the real identities of data providers during the data sharing process, an anonymous certificate-based data sharing policy is proposed. At last, two novel non-interactive zero-knowledge proofs are designed to hide the identities of qualified data providers while claiming rewards to the system. Through security analysis and performance evaluation, the feasibility and effectiveness of the data sharing scheme are illustrated.

Index Terms—Data sharing, privacy preservation, anonymous certificate, blockchain, zero-knowledge proof

数据共享和数据交易的重要性，以及引出两个数据共享的主要问题。1.激励机制。2.隐私保护

1 Introduction

Facilitated by a variety of intelligent devices and ubiquitous Internet access opportunities, large volume of data is collected and changing the style of human behavior [1]. For example, trajectory data can be analyzed by service providers to infer points of interests in a city and to propose navigation suggestions for users. Social data can be organized together to infer the relationship among users and help content recommendation on social network. In the future, data will play such an important role that more and more intelligent services can be proposed to serve human beings better. Additionally, based on statistics, sensing data including sound, images and other environmental data takes up a large part, which is collected by various sensors on IoT devices [2]. With the help of sensing data, users can make wise decisions and get more intelligent services, such as smart home service and autonomous driving service.

In this context, most of the intelligent services can be constructed with the help of large volume of data analysis. That requires kinds of data from various device owners to be put together via data sharing systems or data trading systems in the IoT [3]. However, data owners in the network hesitate to share their sensing data for two reasons. First, the sensing process will cost much resources, such as storage and communication bandwidth. Without any reward, users will not participate in the crowdsensing process. Thus, how to design a mechanism to motivate data owner to participate in the process is a challenge. Second, the sensing data contains the owner's privacy, such as identity information. Data owners are afraid of exposing their own privacy information to the outside [4]. Therefore, in the sharing process, preserving privacy of data providers is also a serious problem to be solved.

Some previous studies have proposed to solve the incentive problem to motivate data owners to join in the data sharing scenario [5] [6]. For example, paper [5] designed an efficient no-pairing and revocable Attribute Based Encryption(ABE) data sharing scheme, which can improve data security and motivate users to upload data. However, most of the solutions are based on the centralized architecture, where the centralized server will store both the service information and the user data and these data cannot be audited by outside users. Thus, there is a risk that the data stored in the center may be disclosed, if the center server is not absolutely trusted or attacked by malicious users.

Then, a distributed method, blockchain technique, which originates from a Peer-to-Peer system, Bitcoin [7], has been applied in many works for the issues mentioned above. The blockchain technology can be used in data sharing to solve the trust problem in a distributed environment and also provide secure data storage or better distribution[8][9]. However, another problem appears that the transactions on the blockchain can be usually accessed by all the miners in the network, which may disclose the data owner's identity. Together with the shared data, the owner's privacy information may be

- *Chao Wang, Shuo Wang, Xiaoman Cheng, Yunhua He, and Ke Xiao are with the North China University of Technology, Beijing 100144, China. E-mail: wangchao.andy@gmail.com, {shuo_wang98, amycxiaoman}@163.com, {heyunhua, xiaoke}@ncut.edu.cn.*
- *Shujia Fan is with Tsinghua University, Beijing 100190, China. E-mail: shujiafan@126.com.*

Manuscript received 24 June 2021; revised 18 January 2022; accepted 27 January 2022. Date of publication 4 February 2022; date of current version 16 January 2023.
This work was supported in part by the National Natural Science Foundation of China under Grant 61802004 and in part by the Scientific Research Project of Beijing Educational Committee under Grant KM202010009008. (Corresponding author: Shujia Fan.)
Recommended for acceptance by P. Barnaghi.
Digital Object Identifier no. 10.1109/TBDATA.2022.3148181

exposed to the outside too. Even with Zcash, a decentralized anonymous payment solution, the transaction data/sharing data and electronic account will be exposed to the data requestor and broker, although those will not be exposed to the miners on the blockchain [10]. So far, there is no targeted research to simultaneously solve the two problems, namely the audit of the center service provider and the privacy preserving for data owners, in the data sharing scheme in the IoT.

Motivated by the challenges, we propose a blockchain-powered data sharing scheme with privacy preservation mechanism. First, a blockchain-based data sharing framework with a proper motivation strategy is proposed. Any data requestor can publish its announcement in the data sharing framework, and data owners can reply to the requestor with sensing data for certain rewards. Important information such as data evaluation and transaction information is stored on the chain, which can be audited by all the users. Then, in order to hide the real identities of participants, an anonymous certificate-based data sharing policy is utilized. The data owners can share data with their anonymous certificates without exposing their real identities. At last, while claiming rewards, data providers have to use their electronic accounts, which can be linked to their real identities. Thus, two novel non-interactive zero-knowledge proofs are designed to verify the identities of qualified data owners without exposing their identities. Following the above steps, a data owner can share its sensing data for rewards without privacy leakage. The main contributions of this paper are listed as follows.

- A blockchain-based data sharing scheme is proposed in this paper. The data evaluation and reward information about shared data is stored on the blockchains, which can be effectively audited by all the participants in the network.
- An anonymous communication policy based on anonymous certificate is proposed. Real identities of users are loosely coupled with their communication behavior, so that the anonymity of user is preserved in the data sharing process.
- Non-interactive zero-knowledge proof is used to break the connections between data providers and their shared data in our mechanism. Any malicious user is incapable of linking the shared data content to its provider.
- We conduct a detailed security and privacy analysis to prove that our mechanism is trustful. Besides, we evaluate the feasibility and effectiveness of the data sharing scheme with a simulation.

The remainder of this paper is organized as follows. Section 2 introduces the related work about data sharing. In Section 3, the system model, threat model, and design goals are presented. Section 4 offers the details of our scheme. Furthermore, comparative privacy and security analysis are carried out in Section 5. Then, we present the evaluation result in Section 6, with Section 7 concluding this article.

2 Related Work

2.1 Blockchain-Based Data Sharing Scheme

In order to design a proper motivation strategy in data sharing, some researches regard data sharing as a transaction, so as to optimize the overall profit of data sharing and encourage data owner to participate in data sharing. In addition, in order to protect the interests from data providers, many researchers have also introduced blockchain technology to monitor participants in data sharing. Paper [11] proposes a blockchain-based data trading protocol involving data providers, consumers, and a Smart Contract, which can reduce the loss caused by fraud. This protocol stipulates that each participant periodically reports their transaction information to the Smart Contract, and the Contract will be able to use the reports to resolve any disputes to safeguard the interests of data providers. In order to solve illegal resell data problem in data sharing, Huang *et al.* [12] design a profit sharing mechanism to encourage reseller to forward and resell data. Besides, a Smart Contract-based protocol is also proposed to supervise data consumption and benefit distribution. Hu *et al.* [13] propose a Smart Contract-based price bargaining mechanism to resolve price disputes between data providers and data buyers. They also propose three detailed data quality evaluation mechanisms to evaluate the quality of transaction data and distribute the revenue of the data according to the result of data quality evaluation.

In addition, as for some relatively large-scale data, researchers have proposed some data auction models for data sharing. Chen *et al.* [14] regard reverse auction as a process of data sharing, in which the cloud server acts as an auctioneer that purchases data from data owners. And the auction platform uses data quality-driven auction model to decide winners and payments and reaches the goal of maximizing social welfare. Gao *et al.* [15] analyse a basic privacy-preserving auction scheme(PPAS) that achieves privacy protected data sharing by encrypting buyer's bids in the third-parity auction platform. And they propose an Enhanced Privacy Preserved Auction Scheme(EPPAS), which can also improve the efficiency of the auction process.

In general, in order to encourage data owners to participate in data sharing, researchers have designed two sharing models: data transaction and data auction. The introduction of blockchain technology can supervise entities participating in data sharing, especially centralized institutions with huge powers. Although the blockchain can make transactions subject to supervision and protect the interests of data providers, it will also expose users' privacy due to the publicity of transaction records. User privacy leakage can be another factor that prevents data owners from participating in data sharing.

区块链虽然可以让交易受到监管，但是交易记录的公开会暴露用户的隐私。

2.2 Data Sharing Scheme With Privacy Preservation

Many researchers have paid attention to the issue of privacy exposure in data sharing, and they have proposed various data sharing schemes with privacy protection functions. Lu *et al.* [16] propose a novel scheme based on federated learning and Deep Reinforcement Learning to relieve transmission load and address privacy issues of data providers. This asynchronous federated learning mechanism for learning models from the edge data minimizes the total cost by selecting participating nodes and further improves the efficiency of federated learning. In order to verify the authenticity of collected data and protect the privacy of users, paper [17] uses homomorphic encryption to construct a ciphertext

space, which can guarantee the normal operation of data services in the case of data encryption. And based on this ciphertext space, they propose an identity-based signature scheme and a two-layer batch signature verification scheme. In paper [18], the authors design a blockchain-based data sharing framework for distributed multiple parties. And they use privacy-protected federated learning to protect data privacy by sharing data models instead of reveling actual data. Deng *et al.* [19] propose a flexible privacy-preserving data sharing(FPDS) model that uses identity-based encryption(IBE) to achieve secure data storage and access. In this scheme, the data owner generates delegation credentials for the data requestor to achieve access control.

In conclusion, to protect privacy in data sharing, many researchers have adopted a variety of methods (e.g., encryption-based methods and deep learning methods) and most of them focus on the data usage process, access control and the data itself. However, the researchers did not study well the privacy leakage problem in the process of acquiring rewards, which can also expose privacy information.

2.3 Our Solution

Although these existing works explore the potential of applying the blockchain technology and some privacy preserving techniques for data sharing, some challenges still exist. The current research on the blockchain-based data sharing with privacy protection cannot absolutely protect data owner's identity privacy. In particular, when the data provider claims reward, the real account address is required to obtain the reward. Our proposed framework considers the privacy leakage problem in the communication process and the rewarding process at the same time, with the techniques including anonymous communication and zero-knowledge proof. Furthermore, to the best of our knowledge, our work is the first to combine privacy preservation into the claiming reward process, leading to desirable identity privacy-preserving during data sharing.

3 System Overview

In this section, the system model of data sharing in the IoT is introduced with the main components described. Based on the threat model analyzed in part 3.2, the design goals of the proposed solutions are also presented.

3.1 System Model

The same as most of IoT architectures [20], such as Internet of Vehicles(IoV) [21] and Industrial Internet of Things(IIoT) [22], the system model is mainly composed of Service Providers(SP), Certificate Authority (CA) and IoT terminals. As we focus on the scenario of data sharing in the IoT, the service provider should act as a Data Sharing Service Provider (DSSP). In the context, IoT terminals/devices can be divided into two parts according to their roles, namely Data Requestors and Data Providers. Thus, the system model can be illustrated as Fig. 1. The following part will list the responsibility of each entity.

Data Requestor: The data requestors publish data request announcements in the network to ask for certain kind of data from other IoT devices. At the same time, the reward is provided along with the announcement and then forwarded to the other terminals as incentives.

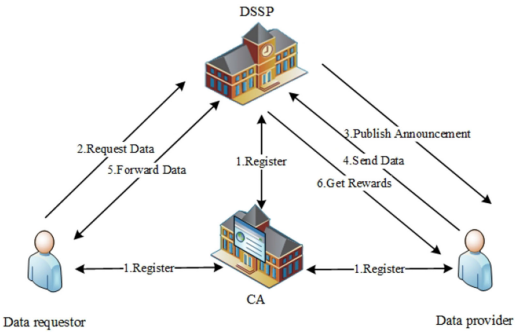


Fig. 1. The system model of data sharing in the IoT.

Data Provider: IoT terminals usually have a variety of sensors for collecting data. Surrounding environment status can be perceived and collected by data providers, and then shared to the data requestor for rewards.

Certificate Authority (CA): Like other data sharing mechanisms based on public key infrastructure(PKI) in the IoT, the responsibilities of the CA include device registration, and certificate publishing and revoking. In our proposed scheme, we use the CA to achieve security communication. (1) Terminal identities are verified before accessing to the data sharing architecture, so that the probability of malicious terminal entering the network is decreased. (2) Once a device has successfully registered in the system, the CA should generate a legal certificate to the device and record the identity of the device in the database. Each terminal in the network can verify the identities of other terminals with the help of the CA. (3) The CA manages all the devices in the network and once a device violates the rules, the CA can revoke its certificate.

Data Sharing Service Provider (DSSP): The DSSP is the core of the system and provides a data sharing platform. It is responsible for issuing data request announcements, forwarding collected data to requestors and helping requestors award data providers. Once the DSSP receives data request information and relative rewards from data requestors, it publishes an announcement based on the request in the network. Besides, the DSSP forwards the data collected by the providers to the corresponding data requestor and pays for the data providers who shared qualified data.

Based on the above entities in the network, the edges with labels (1) - (6) in Fig. 1 illustrate the information flow of the data-sharing scenario.

(1) All the terminals in the network and the DSSP register on the CA and get legal certificates. Next their communications in the network are permitted and can be verified.

(2) Data requestor generates a data request and forwards it with corresponding rewards to the DSSP. The amount of the reward is determined by the requestor, and transfer with electronic account is adopted.

(3) The DSSP publishes the data request announcement with the reward information to other terminals in the network. Any terminal can act as a data provider to finish the data collection task for rewards.

(4) Data providers collect relevant data by sensors and send back to the DSSP.
(5) The DSSP forwards these data to the corresponding requestor. Next, the data requestor evaluates each piece of data to verify its availability and gives the DSSP feedback.
(6) Based on the feedback of the data requestor, the DSSP makes award to qualified data requestors. Similarly, the transaction method for rewards is still electronic transfer.

Once the above steps are finished, the requestors can obtain their necessary data and the providers can get their rewards for their work.

CA是可信的，DSSP是不可信的。DR可能泄露数据，DP只关心在隐私保护的前提下获得更多的钱。

3.2 Threat Model

In this paper, we assume that the CA could be trusted that they will not expose the identities of users to the outside. For the other entities, they have the following behavior characteristics. First, data requestors follow the steps of data sharing protocols, but they are not responsible for preserving other terminals' privacy and they may leak the data content to any one. Second, the data providers just concern about having more rewards while the privacy is preserved in the data sharing process. At last, the DSSP cannot be trusted, it will follow the data sharing protocols but may have some inappropriate behavior for its own benefit with other's benefit loss. Therefore, based on the above assumption and description, the architecture faces the following two threats.

3.2.1 Benefit Encroachment Attack

(1) All the data shared should be forwarded by the DSSP. If it is transferred in plaintext, the DSSP will have a backup of all the data without requestor's permission and any cost, which will encroach the benefits of requestors.
(2) The data requestor only evaluates data and announces the qualified providers. All the rewards for data sharing should be distributed by the DSSP. Under the absence of supervision, the awards allocated to the data providers may be deducted by the DSSP.
(3) For some purposes, a qualified provider may ask for rewards more than once related to one piece of data. This may cause other legal providers unable to get their deserved rewards.

1.DSSP可能备份数据
2.DSSP可能恶意扣钱
3.合格的数据所有者可能多次要求奖励？？

3.2.2 Privacy Leakage

Privacy leakage in this paper stands for the leakage of the relationship between the data provider and its shared data content, which is revealed from two aspects.

(1) Each piece of shared data should be signed by the provider. As the signature can be linked to a terminal in the network, the data content will expose the terminal's privacy to the DSSP and the requestor. Beside, according to the certificate of each provider, different shared data from the same provider to different request can also be linked together, which will expose the terminal's privacy. Even the data is encrypted, the requestor can still get the provider's privacy after its decryption.

这里的隐私泄露是指身份隐私。
1.同一终端对于不同请求容易被链接。
2.转账的时候会泄露DP的身份。

(2) The electronic transfer is used as the reward transaction method in the system and the electronic account number must be provided when conducting electronic transfer. As the electronic account number stands for the unique terminal in the network and generally is not changed, the DSSP can also get the identity information of data providers. Even though the data shared is encrypted, if the DSSP and the requestor collude or the requestor leaks the data content or data characteristics to the DSSP, the DSSP can get the map relationship between the provider's identity and its shared content after analysis.

3.3 Design Goals

In this paper, we propose a data sharing framework with privacy preservation and benefit conservation. In terms of conserving the benefits of terminals(requestors and providers), we desire our framework to achieve the following two purposes.

Data Confidentiality: The data shared to the requestors should be encrypted first to keep the data confidential. Without the requestor's permission, only can the requestor and the provider know the content of the shared data and the DSSP cannot decrypt the data. If the requestor and the DSSP conclude together, it is the same as the scenario that the requestor transfers requested data to the DSSP after decryption. Here, we cannot stop the requestor as the requestor has the right to tackle its received data. This is not the case of violating Data Confidentiality.

评估结果应该接受检验

Unforgeability: The unforgeability is reflected in two aspects. The data evaluation results and the transaction data should be recorded and audited by the public. On one hand, this measure can protect the DSSP from deducting rewards for providers. On the other hand, this is to prevent the malicious provider from asking for rewards twice for the same piece of data.

In terms of privacy preserving, we desire our framework to achieve the following two targets.

Anonymity: When data providers submit data to the DSSP, they should use anonymous certificate to communicate with the DSSP. The real identity cannot be exposed to the DSSP and only the CA can verify the real identity of the provider. Additionally, a terminal should use different anonymous certificates for each data request session to protect the others from linking all the data belonging to the same provider together according to the certificate.

Unlinkability: Since the electronic account number also stands for the terminal's real identity and must be provided while claiming rewards, neither of the data requestor and the DSSP can link the electronic account of the provider to its shared data content, even though the DSSP and the requestor collude together.

4 The Proposed Scheme

In this section, we first propose a data sharing framework with privacy preserving and then describe the details of the proposed scheme. For easier reading, we also give a list of notations to be used in our scheme in Table 1.

TABLE 1
Notations Used in Our Scheme

Notations	Definition
G_1, G_2, G_T	three multiplicative cyclic groups
q	a large prime
g_1, g_2	generators of G_1 and G_2 respectively
$e(.,.)$	a non-degradable bilinear mapping
s_1, s_2	the master secret keys
$H, H^{'}, \hat{H}$	three secure cryptographic hash functions
β	the short time verification key
RID_i	real identity of user i
PID_i	pseudo identity of user i
AK	the authorization key
r_k	the temporary short time private key
Y_k	the temporary short time public key
$(r_d, Y_d = g_1^{r_d})$	the DSSP's private key and public key
$Cert_k$	the temporary short time anonymous certificate
X_1, X_2, X_3	$X_1 = g^{x_1}, X_2 = g^{x_2}, X_3 = g^{x_3} and\ x_1, x_2, x_3 \in Z_q$
θ	the dynamic verification key
Θ, Q	two sets for storage
$cred$	the anonymous credential of user

4.1 System Design

Based on the previous analysis, we introduce anonymous certificate, blockchain technology and zero-knowledge proof technique to solve the issues. The anonymous certificate is used to protect the terminal's real identity information during data transmissions. Thus, the function of the CA is extended to assist the registered terminals to generate anonymous certificates. The blockchain is utilized in our scheme to record the evaluation information of shared data and transaction information of rewards. This is to supervise the behavior of the DSSP and prevent it from forging data and deducting rewards. In addition, the zero-knowledge proof is used to protect the terminal's identity information in the process of claiming rewards. Overall, there are three major steps in the proposed scheme, as shown in Fig. 2, namely system initialization(including A1-A2), anonymous certificate-based data sharing(including B1-B5) and zero-knowledge-proof-based rewarding (including C1-C4).

System Initialization. ($A.1$) Terminals and the DSSP in the data sharing scenario send the required information to the CA for registration. ($A.2$) The CA publishes the communication parameters, assist terminals to generate anonymous certificates.

Anonymous Certificate-Based Data Sharing. ($B.1$) A data requestor sends data requirement and the corresponding rewards to the DSSP. ($B.2$) The DSSP forms a data request announcement and distributes it to the public. ($B.3$) Data providers collect sensing data and use pseudo identities to share the sensing data which is encrypted in advance to the DSSP. ($B.4$) The DSSP checks the legality of the received data with the providers' anonymous certificates. The legal data is forwarded to the data requestor, without exposing the owner's information. ($B.5$) The data requestor evaluates the received data and records the evaluation results on the data evaluation chain.

Zero-Knowledge-Proof-Based Rewarding. ($C.1$) The DSSP initializes public parameters for qualified providers to claim rewards. ($C.2$) Each data provider checks whether its shared data can be awarded. If so, the qualified data provider uses its pseudo identity, with which it sent the data during the data provision phase, to ask for rewards from the DSSP. Then, the DSSP sends anonymous credentials to qualified data providers. ($C.3$) The data provider uses anonymous credential and rewarding token to accept rewards with its electronic account. ($C.4$) After the DSSP distributes the reward to the qualified provider, it records the transaction information on the transaction chain.

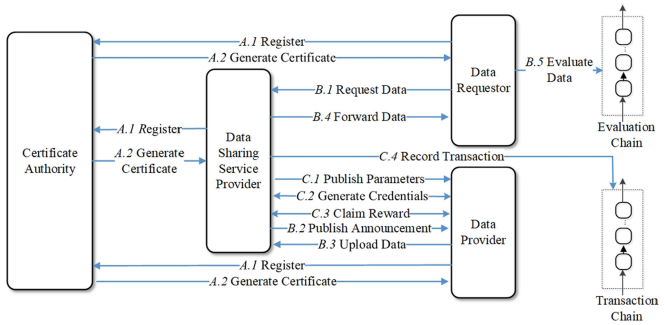


Fig. 2. System model of the proposed scheme.

4.2 System Initialization

为保护信息的真实性，必须签名

A.1 Terminals and the DSSP Registration

In our proposed scheme, it is necessary to ensure the authenticity of messages while a terminal uploads data. Therefore, any terminal that intends to participate in data sharing must register to the CA to obtain a legal identity. In the registration process, terminals submit the required information to the CA. If passing this registration, the terminals are considered as legal participants. The same as terminals, the DSSP also needs to register in the CA to get a legal identity.

A.2 Certificate Generation

In order to achieve anonymous data sharing, we propose an anonymous communication mechanism. Each terminal can use an anonymous certificate obtained from the CA to communicate with the others, which can ensure that the real identity of the terminal is hidden. Besides, the anonymous certificate can guarantee the authenticity of the publisher's identity and the integrity of the message. At last, the CA can still revoke the user's certificate, even if the user uses an anonymous certificate. The following is the specific steps for anonymous certificate genarations.

(1) Anonymous Certificate for Terminal

(a) The CA generates three multiplicative cyclic groups G_1, G_2 and G_T with the same order q, where q is a large prime. g_1 and g_2 are the generators of G_1 and G_2 respectively. $e(.,.)$ denotes a bilinear map such that $e: G_1 \times G_2 \longrightarrow G_T$. Two numbers $s_1, s_2 \in Z_q^*$ are chosen by the CA as the master secret keys and calculates $P_1 = g_1^{s_1}$ and $P_2 = g_1^{s_2}$, where $Z_q^* = [1, \ldots, q-1]$. Next, the CA chooses a random number $t \in Z_q^*$ and computes $\beta = g_1^t$ as the short time verification key. Here, the short time verification key changes according to the system requirement (i.e., every day, every hour, etc) and the terminal can ask for the corresponding β based on current timestamp. Additionally, the CA also chooses a secure cryptographic hash function $H: \{0,1\} \longrightarrow Z_q^*$. Then, the tuple $\{q, e, g_1, g_2, G_1, G_2, G_T, P_1, P_2, H, \beta\}$ is published as the public communication parameters to the outside.

(b) If a terminal i has successfully registered, the CA assigns the necessary keys to the terminal. The CA creates a linkage between its real identity(RID_i) and its pseudo

identity(PID_i). Here, RID_i is the original id of terminal i, which may be its device number or IMEI number standing for its social identity. For the PID_i, the CA randomly selects a number $a_i \in Z_q^*$ and computes $PID_i = g_1^{a_i+s_1} mod\ q$. Additionally, a random number $b_i \in Z_q^*$ is chosen by the CA and compute $T_i = g_1^{\frac{1}{b_i+s_1+s_2}}$. At last, the CA stores $(RID_i, PID_i, T_i^{s_2})$ locally and forwards an authorization key $AK_i = (PID_i, T_i, E_i)$ where $E_i = g_1^{-a_i} mod\ q$ to the terminal for communication in the future.

(c) When the terminal intends to participate in data sharing, it uses the AK to generate the anonymous certificate. Each terminal randomly chooses some numbers $r_1, r_2, \ldots, r_l \in Z_n^*(l \leqslant n)$ as the temporary private keys and calculates the corresponding temporary public keys $Y_k = g_2^{r_k}(k \in \{1, 2, \ldots, l\})$. Next, the terminal chooses some random numbers $\mu, c_1, c_2 \in Z_q^*$ and calculates $PID_i^{'}, E_i^{'}, \varphi_1, \varphi_2, \lambda, \lambda_1, \lambda_2, \delta_1, \delta_2$ as follows:

$$
\begin{aligned}
PID_i^{'} &= e(PID_i, \beta) \\
E_i^{'} &= e(E_i, \beta) \\
\varphi_1 &= P_2^{\mu} \\
\varphi_2 &= T_i \cdot P_1^{\mu} \\
\lambda &= (\mu + r_k) mod\ q \\
\lambda_1 &= \varphi_1^{\mu+c_1} \\
\lambda_2 &= \frac{\varphi_1^{\mu+c_1}}{\varphi_2^{\mu+c_2}} \\
\delta_1 &= (r_k - c_1) mod\ q \\
\delta_2 &= (r_k - c_2) mod\ q
\end{aligned}
$$

(d) Finally, the anonymous certificate is denoted as $Cert_k = \{PID_i^{'} \parallel Y_k \parallel E_i^{'} \parallel \varphi_1 \parallel \varphi_2 \parallel \lambda \parallel \delta_1 \parallel \delta_2 \parallel c \parallel timestamp\}$, where $c = H(PID_i^{'} \parallel e(P_1, \beta) \parallel e(P_2, \beta) \parallel E_i^{'} \parallel \varphi_1 \parallel \varphi_2 \parallel Y_k \parallel \lambda_1 \parallel \lambda_2)$ and $timestamp$ is the time certificate generated.

(2) Certificate for the DSSP

The DSSP also requires a key pair, which can help other users to verify the authenticity of the information sent by the DSSP. The DSSP first selects a random numbers $r_d \in Z_q^*$ and computes $Y_d = g_1^{r_d}$, where r_d is the private key and the Y_d is the public key. Next, the public key Y_d is sent to the CA for legal certificate, which can be published in the network.

4.3 Anonymous Certificate-Based Data Sharing

B.1 Data Request

In the scenario, IoT terminals can request the specific data they need. Once a terminal needs data, it sends a data request message $ReqMeg = (req \parallel sig \parallel Cert_k)$ and the corresponding rewards as deposit to the DSSP. Here, req includes PID of the requestor, data requirement, the amount of rewards and a public key, where the public key is used by the provider to encrypt sharing data and PID is used to verify the identity of the requestor to prevent the public key from being tampered. Besides, in order to prevent the DSSP from maliciously changing the content of req, $ReqMeg$ also includes a signature $sig = g_1^{\frac{1}{r_k+H(req)}}$ and a certificate $Cert_k$. At last, it is assumed that the DSSP will evenly distribute the rewards to the qualified data providers on its behalf at the end of data sharing.

B.2 Publish Request Announcement

After receiving the deposit, the DSSP forms a data request announcement $anno$ which includes the announcement id and the data request $ReqMeg$ from the requestor. The DSSP produces a signature $sig = g_1^{\frac{1}{r_d+H(anno)}}$ using its private key r_d and the announcement $anno$. Then, the DSSP distributes the announcement message $AnnMsg = (anno \parallel sig \parallel Y_d)$ to the public.

B.3 Data Collection and Upload

Once the data provider collects sensing data according to the request announcement, it encrypts the data using the public key included in the announcement. The provider can utilize the $ReqMeg$ information in the announcement, namely (req, sig, $Cert_k$) to verify the authenticity of the public key in the req. This process is described in the following Section 5.1.

In order to guarantee the integrity of the encrypted data D and the anonymity of the provider, the data provider generates a temporary anonymous signature $sig = g_1^{\frac{1}{r_k+H(D)}}$ with the temporary private key r_k(generated in part A.2). For each round of data providing, the provider will change the temporary private key, so that the public key changes for each round too.

Then, the provider shares the data D via the anonymous message $msg = (D \parallel sig \parallel Cert_k)$ to the DSSP. Here, $Cert_k$ is the anonymous certificate generated in Part A.2. In addition, in order to claim rewards in the future, the data provider will also store the current temporary key pair and anonymous certificate.

B.4 Data Verification and Forwarding

(a) The DSSP verifies the received data before forwarding it to the requestor. To ensure the provider of the shared data is legal, the DSSP first calculates the following values:

意思是，验证者需要自行计算兰姆达1和2，

$$
\begin{aligned}
N_i &= E_i^{'} \times PID_i^{'} \\
\lambda_1^{'} &= \frac{\varphi_1^{\lambda}}{\varphi_1^{\delta_1}} \\
\lambda_2^{'} &= \frac{\varphi_1^{\lambda} \cdot \varphi_2^{\delta_2}}{\varphi_1^{\delta_1} \cdot \varphi_2^{\lambda}}
\end{aligned}
$$

Once the DSSP obtains the corresponding β according to the $timestamp$, it calculates $c^{'} = H(PID_i^{'} \parallel N_i \parallel e(P_2, \beta) \parallel E_i^{'} \parallel \varphi_1 \parallel \varphi_2 \parallel Y_k \parallel \lambda_1^{'} \parallel \lambda_2^{'})$ and judges whether $c = c^{'}$. If the equation holds, the DSSP accepts the anonymous certificate $Cert_k$. Otherwise, the DSSP discards the data.

(b)Next, the DSSP uses the following equation to verify the integrity of the data.

$$e(sig, Y_k \cdot g_2^{H(D)}) = e(g_1, g_2)$$

If the equation holds, the DSSP will forward the encrypted data to the requestor, without exposing the provider's information such as the anonymous certificate. Otherwise, the data will be discarded. In addition, the DSSP will record the digest of the encrypted data D and its corresponding anonymous certificate to verify the legality of the provider in the following awarding phase(Section 4.4 C.2).

B.5 Data Evaluation

In this paper, we assume that all the data shared is evaluated as qualified data (denoted as True) or unqualified data

上链的评估结果包括数据片段摘要和结果。

(denoted as False), and all the qualified data providers will share the total rewards for the request evenly. When the data requestor receives the data from the DSSP, it first decrypts the data with its private key and evaluates it. Once all the data related to the request has been evaluated, the evaluation list(including the digest of the piece of encrypted data, the evaluation result) is recorded on the evaluation chain by the requestor, which is described in the following Section 4.5.

The DSSP and all the providers can check the evaluation result via the evaluation chain. If the data is evaluated as qualified data, the link between the digest of the data and its corresponding anonymous certificate is established locally at the DSSP. Then, the DSSP can infer whether an anonymous certificate is eligible for the rewards based on the digest set shown on the evaluation chain.

4.4 Zero-Knowledge-Proof-Based Rewarding

In this section, we will describe how to use zero-knowledge proof to receive rewards without exposing user account privacy. The zero-knowledge proof is a method by which one party (the prover) can prove to another party (the verifier) that they know a value x, without conveying any information apart from the fact that they know the value x. In our proposed scheme, the zero-knowledge proof is used to cut off the connection between transaction account information and data. When receiving the award, the provider proves to the DSSP that he has sent qualified data without disclosing any information. Then, the provider can provide account information to receive rewards without worrying that the DSSP will associate account information with the data provided.

C.1 Initialization of Awarding Parameters

In this phase, the DSSP will initialize public parameters for awarding. As we mentioned above, G_1 and G_T are two multiplicative cyclic groups with generators g_1 and g_T respectively. The DSSP first selects two cryptographic hash functions $H^{'}$ and $\hat{H}$, such that $H^{'}: Z_q \longrightarrow Z_q$ and $\hat{H}: \{0,1\}^{*} \longrightarrow Z_q$. Second, it selects $\theta \in Z_q$ as a dynamic verification key, with its value changing according to the request announcement id. Third, it randomly chooses three numbers $x_1, x_2, x_3 \in Z_q$ and computes $X_1 = g_1^{x_1}, X_2 = g_1^{x_2}$ and $X_3 = g_1^{x_3}$. At last, the DSSP publishes the awarding parameters $\{G_1, G_T, q, g_1, g_T, e, X_1, X_2, X_3, H^{'}, \hat{H}, \theta, Y_d\}$ for terminal claiming rewards. In addition, the DSSP generates two empty sets $\Theta = \{\emptyset\}$ and $Q = \{\emptyset\}$ locally.

C.2 Credential Generation

(a) As we mentioned above, the data evaluation result for each request is stored on the evaluation chain by the data requestor. Thus, the providers can obtain awarding information by searching on the evaluation chain. If the tag of data someone provides is True, it can get rewards.

(b) The data provider who is eligible for rewards chooses $(z_1, z_2) \in Z_q^2$ to calculate $M = X_2^{z_1} X_3^{z_2}$. Next, it sends $(M \parallel H^{'}(z_1) \parallel sig \parallel Cert_k \parallel announcement\ id)$ to the DSSP with the anonymous certificate used in the data provision phase and the corresponding key pair (r_k, Y_k) generating sig.

(c) Once the DSSP received that message, based on the anonymous certificate inside, it first judges whether the provider is eligible for rewards and whether it has already received the reward. If the provider is qualified and has not yet received the reward, then the DSSP judges whether $H^{'}(z_1)$ is in Θ. If not, the DSSP adds $H^{'}(z_1)$ into Θ and notifies the provider. Otherwise, the DSSP notifies the provider to change z_1. This is to prevent legitimate providers in the same data request from obtaining rewards with the same z_1. So far, the DSSP has already verified the provider's eligibility for rewards.

Then, the data provider still uses the anonymous certificate to perform a zero-knowledge proof with the DSSP. Here, the provider acts as a prover and the DSSP as a verifier in zero-knowledge proof. Via zero-knowledge proof, the provider can use an anonymous credential to stand for itself and claim rewards without utilizing original anonymous certificate. The details of the zero-knowledge proof are described as follows and more can be found in [23].

Proof. I $ZkPoK\{(z_1, z_2)|M = X_2^{z_1} X_3^{z_2}\}$

Prover:

(1) Choose $z_\alpha, z_\beta \in Z_q$, calculate $\sigma = X_2^{z_\alpha} X_3^{z_\beta}$

(2) Set o=$\hat{H}(X_2, X_3, M, \sigma)$

(3) Send $(\sigma, M, \hat{z_\alpha}$=$z_1 o + z_\alpha, \hat{z_\beta}$=$z_2 o + z_\beta)$*to the verifier*

Verifier:

(1) Calculate o=$\hat{H}(X_2, X_3, M, \sigma)$

(2) Check that $M^o \sigma = X_2^{\hat{z_\alpha}} X_3^{\hat{z_\beta}}$

(d) If the proof is successful, it can be inferred that the provider has (z_1, z_2) to stand for itself without exposing the content to the DSSP. Next, the DSSP sends the data provider a tuple (γ, W), where $\gamma \in Z_q$ and $W = (X_1 M)^{\frac{1}{\gamma + r_d + \theta}}$. After the provider verifying $e(W, Y_d g_1^{\gamma+\theta}) = e(X_1 M, g_1)$, a credential $cred = (W, \gamma, z_1, z_2)$ is stored locally by the data provider. In the future, the data provider will prove to others it has the credential $cred$ to claim rewards. In addition, in order to ensure that legitimate users can receive rewards and can only obtain for one time, once the DSSP sends the tuple (γ, W) to the provider, it will mark the anonymous certificate for qualification verification by the provider as used to prevent from claiming awards again with the same certificate.

C.3 Rewards Claim

The qualified data provider should send the announcement id to the DSSP for rewards together with its electronic account information. The DSSP fixes the awarding amount according to the announcement id. Then the DSSP and the provider will perform another zero-knowledge proof to verify the availability of the credential held by the provider. Here, the provider acts as a prover and the DSSP as a verifier in the zero-knowledge proof. The provider will prove it has the credential without showing the credential to the DSSP. The details of the zero-knowledge proof are described as follows.

Proof. II $ZkPoK\{(W, \gamma, z_1, z_2)|W^{\gamma+r_d+\theta} = X_1 X_2^{z_1} X_3^{z_2} \bigwedge R = g_1^{\frac{1}{z_1+\theta}}\}$

According to the paper [24], the proof can be transformed and rewritten as

$ZkPoK\{(\gamma, z_1, z_2, \eta_1, \eta_2, \xi_1, \xi_2)|W_1 = X_2^{\eta_1} X_3^{\eta_2} \wedge 1_{\mathbb{G}} = W_1^{-\gamma} X_2^{\xi_1} X_3^{\xi_2} \wedge R^{z_1} = g_1 R^{-\theta} \wedge \frac{e(W_2, Y_d g_1^{\theta})}{e(X_1, g_1)} = e(W_2, g_1)^{-\gamma} e(X_2, g_1)^{z_1}\ \ e(X_3, Y_d)^{\eta_1} e(X_3, g_1^{\theta})^{\eta_1} e(X_3, g_1)^{z_2 +}\ \xi_1\}$

where $\eta_1, \eta_2 \in Z_q$, $W_2 = W X_3^{\eta_1}$, ξ_1=$\eta_1 \gamma$, and $\xi_2 = \eta_2 \gamma$

Prover:

(1) Choose $I_\gamma, I_{z_1}, I_{z_2}, I_{\eta_1}, I_{\eta_2}, I_{\xi_1}, I_{\xi_2} \in Z_q$, calculate

$R = g_1^{\frac{1}{z_1+\theta}}, \sigma_1 = X_2^{I_{\eta_1}} X_3^{I_{\eta_2}}, \sigma_2 = W_1^{-I_\gamma} X_2^{I_{\xi_1}} X_3^{I_{\xi_2}}, \sigma_3 = R^{I_{z_1}}, \sigma_4 = e(W_2, g_1)^{-I_\gamma} e(X_2, g_1)^{I_{z_1}} e(X_3, Y_d)^{I_{\eta_1}} e(X_3, g_1^{\theta})^{I_{\eta_1}} e(X_3, g_1)^{I_{z_2} + I_{\xi_1}}$

(2) set $o = \hat{H}(X_1, X_2, X_3, W_1, W_2, R, \sigma_1, \sigma_2, \sigma_3, \sigma_4)$

(3) send $(\sigma_1, \sigma_2, \sigma_3, \sigma_4, W_1, W_2, R, \hat{I_\gamma} = \gamma o + I_\gamma, \hat{I_{z_1}} = z_1 o + I_{z_1}, \hat{I_{z_2}} = z_2 o + I_{z_2}, \hat{I_{\eta_1}} = \eta_1 o + I_{\eta_1}, \hat{I_{\eta_2}} = \eta_2 o + I_{\eta_2}, \hat{I_{\xi_1}} = \xi_1 o + I_{\xi_1}, \hat{I_{\xi_2}} = \xi_2 o + I_{\xi_2})$ to the verifier.

Verifier:

(1) calculate $o = \hat{H}(X_1, X_2, X_3, W_1, W_2, R, \sigma_1, \sigma_2, \sigma_3, \sigma_4$

(2) check that $W_1^o \sigma_1 = X_2^{\hat{I_{\eta_1}}} X_3^{\hat{I_{\eta_2}}}$, $1_{\mathbb{G}}^o \sigma_2 = W_1^{-\hat{I_\gamma}} X_2^{\hat{I_{\xi_1}}} X_3^{\hat{I_{\xi_2}}}$, $(g_1 R^{-\theta})^o \sigma_3 = R^{\hat{I_{z_1}}}$, $(\frac{e(W_2, Y_d g_1^\theta)}{e(X_1, g_1)})^o \sigma_4 = e(W_2, g_1)^{-\hat{I_\gamma}} e(X_2, g_1)^{\hat{I_{z_1}}} e(X_3, Y_d)^{\hat{I_{\eta_1}}} e(X_3, g_1^\theta)^{\hat{I_{\eta_1}}} e(X_3, g_1)^{\hat{I_{z_2}} + \hat{I_{\xi_1}}}$

If the proof is successful and R is not in set Q, the provider can get the reward from the DSSP. Otherwise, the provider cannot be awarded. Here, R is a token that can only be used once. The purpose of checking R in the set Q is to verify whether its rewards have been claimed before as one provider for a data request can only get rewards for once. In addition, the DSSP can also receive some rewards as compensation for forwarding data. When the DSSP distributes rewards to providers, it conserves some rewards in accordance with the pre-agreed ratio in the system(This part is not the focus point in this paper).

C.4 Transaction Record

The transaction information will be recorded on the transaction chain by the DSSP. Any participant can obtain information from the blockchain and audit the transaction information. Some details are described in the following Section 4.5.

4.5 The Structure of Blockchains

As mentioned above, the role of the DSSP is to help data requestors generate request announcements, forward data, and issue rewards. Thus, we introduce blockchains to supervise the behavior of the DSSP and audit the information published by the DSSP, such as the rewards distribution information. In our scheme, the blockchains are used to record evaluation information and transaction information and anyone who participates in data sharing must be registered. Compared with Ethereum, Hyperledger Fabric is more suitable for our scheme. There are three reasons. First, as a typical framework of consortium chain, it has membership management, which is similar to our scheme that all participating users must be registered. Second, it does well on storing public audit data, which is helpful for us to store transaction information and evaluation information. Third, it is based on a modular architecture and has efficient query capabilities, which can improve the efficiency of data sharing.

In addition, our network is not completely trusted, and there may be a small number of abnormal behavior terminals. We employ Practical Byzantine Fault Tolerance (PBFT) protocol to maintain the blockchain. The PBFT can guarantee one-third fault tolerance, which reduces the possibility of incorrect execution results due to potential laziness and dishonest behaviors of nodes.

As we said above, we use two ledgers to record evaluation information and transaction information individually. They have different user chaincodes and their consensus processes are independent with each other. Chaincode, as a smart contract provided by Hyperledger Fabric, has record and query functions [25]. In our scheme, with the chaincode in the evaluation ledger, endorsing nodes only need to verify the identity of the requestor, as the requestor evaluates the data without the information of the provider identity. Then, the evaluation information will be recorded through the PBFT consensus protocol. Unlike the evaluation ledger, records on the transaction ledger require more complex verification with the chaincodes. The transaction information generated by the DSSP needs to be verified, and it should be signed by legal providers during the request. Then, the endorsing nodes will verify the signatures of legitimate providers to ensure the correctness of transaction information. After the endorsement node completes the endorsement, the transaction information can also reach a consensus and be recorded by the PBFT consensus protocol.

The evaluation ledger stores the evaluation information for each piece of data from data providers. The body of the evaluation ledger includes id of the request announcement, digest of encrypted data, evaluation result and digital signature of the record. The evaluation result is denoted as True or False. If the data is evaluated as True, the provider can get rewards; otherwise not.

The transaction ledger is used to store the transaction information for rewards distribution. The difference from the evaluation ledger is that transaction item includes id of the request announcement, amount of the reward, timestamp, signature of data requestors and signature of data providers. The total amount of rewards for each request announcement is determined by the requestor in advance.

5 Security and Privacy Analysis

5.1 Security Analysis

In this section, we will analyze how our proposed scheme protects the benefits of participants.

(1) *Data Confidentiality:* Before the data provider sends data to the DSSP, the data is encrypted by the provider with the public key provided by the requestor. Although the data needs to be forwarded by the DSSP, the data obtained by the DSSP is the encrypted data. Only the data requestor can decrypt the data with its private key, which guarantees the confidentiality of the data.

In addition, the provider can utilize the $ReqMeg$ information in the announcement, namely (req, sig, $Cert_k$) to verify the authenticity of the public key in req. First, our scheme can prevent the DSSP from forging certificate $Cert_k$ in the $ReqMeg$, as the provider can use PID included in the req to calculate $PID_i^{'} = e(PID_i, \beta)$, which is the same as the $PID_i^{'}$ in $Cert_k$ of the $ReqMeg$. Then the public key in $Cert_k$ cannot be replaced with other public keys and the sig in $ReqMeg$ cannot be forged either. At last, the requestor's signature $sig = g_1^{\frac{1}{r_k + \hat{H}(req)}}$ in the $ReqMsg$ can guarantee the integrity of the data request req.

(2) *Unforgeability:* First, the DSSP may maliciously deduct rewards, causing legitimate providers can not to receive corresponding rewards. For this reason, the evaluation chain and transaction chain we introduced are used to record evaluation information and transaction information respectively. Any terminal in the network can check the information on the two chains to supervise the DSSP to distribute rewards to the provider based on the requestor's evaluation. This makes it impossible for the DSSP to forge information to deduct rewards.

Second, for each data sharing session, a malicious provider may ask for rewards twice for the same piece of data. To resist this, our scheme designs a credential generation process and a rewards claim process. The credential generation process mainly verifies the eligibilities of qualified providers and the rewards claim process aims to distribute rewards. In the credential generation process, a qualified provider will use tuple (z_1, z_2) to generate credential $cred = (W, \gamma, z_1, z_2)$. However, before the credential is generated, the provider needs to send $(M \parallel H^{'}(z_1) \parallel sig \parallel Cert_k \parallel announcement\ id)$ to the DSSP so that it can verify the provider's eligibility to receive the award based on anonymous certificate. In addition, the DSSP will mark the anonymous certificate after verification. Thus, the anonymous certificate sent by each provider can only be used for verification once for each data provision. In other words, each legal provider can only generate one credential.

If a dishonest provider makes the forgery attack, the DSSP can figure out that the credential has been issued and cannot be issued again. In the rewards$_1$claim process, the provider claims its reward token $R = g_1^{\frac{z_1}{z_1+\theta}}$, where z_1 is generated by the provider on behalf of its identity and θ is determined by the request announcement, which can guarantee that each provider can only generate one awarding record. If R is maliciously changed by the provider, the *Proof. II* cannot be passed. Similarly, the DSSP deploys a set Q to ensure that the awarding record can only be used once. In general, this ensures that our system can resist forgery of reward information and the rewards will not be stolen.

5.2 Privacy Analysis

We will analyze how our proposed scheme can achieve privacy preservation discussed in Section 3.3.

(1) *Anonymity:* Any participant in the network must have a certificate, which is designed to prove the legitimacy of a terminal's identity. In order to ensure message authenticity without revealing the real identities of terminals, a pseudonym mechanism is proposed to protect the privacy of terminals. The CA creates a pseudo identity PID_i for each user i to generate dynamic pseudonyms $PID_i^{'}$ and temporary anonymous certificates $Cert_k$. Then, while a provider shares data, it will use its anonymous certificate $Cert_k$ with dynamic pseudonyms $PID_i^{'}$ inside standing for its identity. That can hide the provider's real identity to the DSSP and the data requestor, but the identity of the provider can also be checked by the CA with the anonymous certificate $Cert_k$. Especially, when the requestor and the DSSP collude, the requestor only knows the temporary pseudonym of each provider.

Besides, $PID_i^{'} = e(PID_i, \beta)$, $E_i^{'} = e(E_i, \beta)$ and the anonymous certificate $Cert_k = \{Y_k \parallel E_i^{'} \parallel PID_i^{'} \parallel \varphi_1 \parallel \varphi_2 \parallel c \parallel \lambda \parallel \delta_1 \parallel \delta_2 \parallel timestamp\}$ are changed according to the time, so that each terminal has different anonymous certificates in each data sharing session in our proposed scheme. In other words, the data provided by the same provider to different requests cannot be linked together with the anonymous certificate $Cert_k$. With the assumption that the CA is trustful and will not expose the linkage between the real identity RID_i and the dynamic pseudonym PID_i of each participant, the requestor and the DSSP cannot obtain the real identities of data providers.

TABLE 2
Comparison of Functionality

Functionality	paper[19]	paper[26]	paper[27]	Our scheme
Data confidentiality	✓	✓	✓	✓
Unforgeability	✓	✓	✓	✓
Anonymity	✓	✓	×	✓
Unlinkability	×	×	×	✓

(2) *Unlinkability:* For the unlinkability, there are two cases concerning whether the DSSP and the requestor collude.

For the case that the DSSP does not collude with the requestor. The data provider needs to provide account information to get rewards from the DSSP after the provider submits encrypted qualified data to the data requestor. It is obvious that the DSSP can link the provider's account information to the encrypted data. As the DSSP does not collude with the requestor, the DSSP cannot decrypt the data and get the original content of the data from the requestor and the requestor cannot get the provider's identity information from the DSSP. Therefore, in this case neither of the DSSP and the requestor can link provider's account information to its shared data content.

For the case that the DSSP colludes with the requestor. Due to collusion, the DSSP will get decrypted data content from the requestor, which will help the DSSP to link the account information with the original data content. We use two zero-knowledge proofs to prove that the data shared by the provider is qualified without revealing which data the provider transmits.

- In *Proof. I*, the provider can obtain an anonymous credential $cred$ for future rewards claiming without utilizing its original anonymous certificate.
- In *Proof. II*, the DSSP verify the availability of the credential and the unique reward token R to make a reward transaction.

As assumed that all the qualified providers share the rewards for a request equally, the DSSP can only verify whether the provider is legitimate and its account information. However, the DSSP cannot obtain which piece of data is submitted to infer the connection between its account information and its shared data.

5.3 Comparison With Other Works

Additionally, we also compared our scheme with other existing data sharing works concerning privacy preservation. The comparison results are shown in Table 2.

Paper [19] proposed a flexible privacy-preserving data sharing(FPDS) model that uses identity-based encryption (IBE) to achieve secure data storage and access. The data owner can generate delegated credentials to meet the requirements of the data requestor to achieve access control. But the model does not consider the protection of the privacy of the data owner's identity either. Paper [26] proposed a recoverable, revocable, and privacy-preserving edge data sharing scheme. The access control method with adding or revoking attributes for different data users and encryption technology is utilized in the paper. However, the privacy protection of this paper is not considered comprehensively, and it lacks the protection of the identity information. In

TABLE 3
Time Cost of Certificate and Signature Verification

Scheme	For one certificate and signature	For n certificates and signatures
Liu *et al.* [28]	$3T_p+3T_h+3T_m$	$3nT_p+3nT_h+3nT_m$
Shao *et al.* [29]	$3T_p+2T_h+2T_{ep1}$	$(2+n)T_p+2nT_h+2nT_{ep1}$
He *et al.* [30]	$2T_p+6T_h+2T_m$	$2nT_p+6nT_h+2nT_m$
Proposed	$2T_p+4T_{ep1}+T_{ep2}$	$(1+n)T_p+4T_{ep1}+nT_{ep2}$

paper [27], the problem of data sharing by joint consideration on the utility, privacy, and the rationality is formulated. A privacy-preserved data sharing framework with differential privacy is proposed for IIoTs. The proposed scheme aims at disturbing sensitive data in shared data, and does not consider other situations of privacy leakage.

As shown in Table 2, all the proposed schemes satisfy the data confidentially and unforgeability. However, there is no scheme that satisfies the anonymity and unlinkability simultaneously. Our scheme comprehensively considers the various situations of user privacy leakage, and proposes a more comprehensive privacy protection mechanism in data sharing.

6 Performance Evaluation

In this section, we evaluate the performance of the proposed scheme in terms of communication overhead, time and memory cost. Since our solution is particularly proposed for the data sharing process, there are many characteristics previous schemes cannot satisfy (e.g., anonymity, unlinkability, etc.). Hence, we mainly investigate the performance of privacy preservation measures, including anonymous certificate-based data sharing and zero-knowledge-proof-based rewarding. The experiments are conducted as follows: The DSSP is built on a notebook with AMD Ryzen7 4800H CPU@2.90GHz and 16.00 GB memory. The IoT terminal is conducted on a smartphone with Qualcomm Snapdragon 845 CPU, 6.0 GB memory and Android 10.0 operating system.

6.1 Performance of Anonymous Certificate-Based Data Sharing

The DSSP will receive many messages from data providers and data requestors. In order to check the authenticity and the integrity of these messages, the DSSP needs to verify the certificates and the carrying signatures respectively. In this process, the performance of time cost on the certificate and signature verification locates at the core position that users may focus on. The verification delay of our proposed scheme is compared with many existing lightweight anonymous authentication mechanisms in the IoT, including [28], [29] and [30].

Let us consider some major operations to determine the computation cost as follows: T_p denotes the time consumption for performing one pairing operation; T_h denotes the time consumption for performing one hash function and T_m denotes the time consumption for performing one point multiplication. The time consumption for performing exponentiation in G_1 and G_2 are denoted as T_{ep1} and T_{ep2}. As mentioned above, we use the Pairing-Based Cryptography (PBC) library to perform the hash operation, exponential operation, point multiplication and pairing operation. The time consumption obtained is the average value of 100 random simulations.

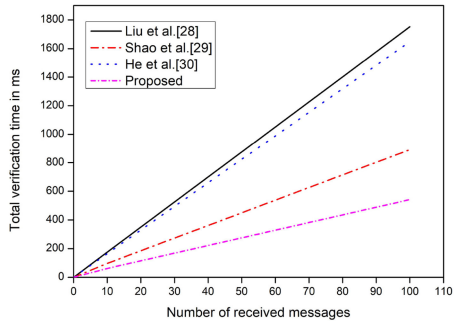


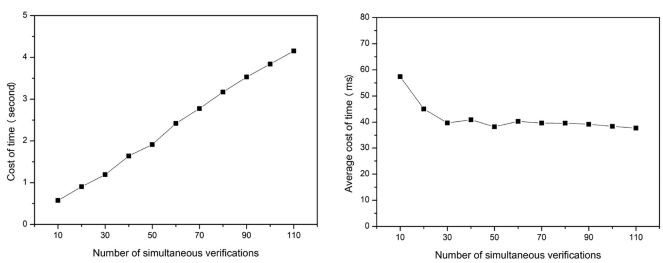
Fig. 3. Comparison on time cost of certificate and signature verification.

In our simulations, the calculation time of each parameters T_p, T_h and T_m are obtained with the results 4.36ms, 1.20ms and 0.28ms respectively. T_{ep1} and T_{ep2} are obtained with the value 1.03ms and 0.976ms respectively. In Table 3, we compare the certificate and signature verification costs of various schemes. As shown in Table 3, our proposed scheme takes lowest computational cost compared to other existing schemes. It can take only $2T_p$, $4T_{ep1}$ and T_{ep2} for verifying one certificate and signature. It takes 13.82ms for our proposed scheme to verify one certificate and signature, while other exiting schemes (Liu *et al.* [28], Shao *et al.* [29] and He *et al.* [30]) take 17.52ms, 17.54ms and 16.48ms respectively. Fig 3 shows the tendency of the time cost with the number of verification messages increasing. As the number of received messages n increases, our proposed scheme is much more efficient than other anonymous authentication schemes.

6.2 Performance of Zero-Knowledge-Proof-Based Rewarding

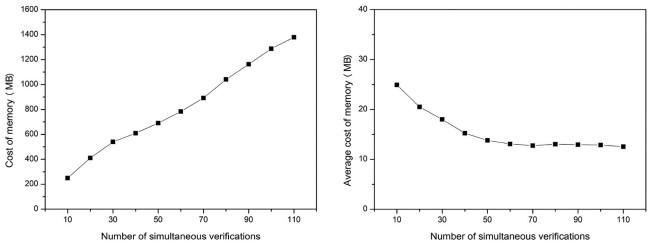
Due to the necessity of privacy preserving, we use zero-knowledge proof in the process of claiming rewards. Zero-knowledge proof requires a prover and a verifier, and the DSSP acts as a verifier with the providers as verifiers in the process of allocating rewards. In order to test the performance of zero-knowledge proof in our system, we simulate the process of the proof on the DSSP server and the Android terminals. In the following, we will test the performance from two aspects, name time consumption and memory consumption.

(1) *Time Consumption:* On the DSSP side, as shown in Fig. 4a, as the number of simultaneous proofs increases, the total time cost also increases almost linearly. However, as shown in Fig. 4b, the average time cost decreases with the



(a) Time consumption (b) Average time consumption

Fig. 4. Time consumption of zero-knowledge proof.



(a) Memory consumption (b) Average memory consumption

Fig. 5. Memory consumption of zero-knowledge proof.

number of simultaneous verifications increasing. At last, the average time cost is tending to be stable. As the DSSP needs to initialize public parameters when it starts a zero-knowledge proof, the average time cost is very high at the beginning. However, the public parameters need to be calculated for just one time, so that the average verification time will decrease with the number of verifications increase and then becomes a constant.

In addition, the IoT terminals usually have limited computing resources. In order to demonstrate whether the two zero-knowledge proofs can be effectively implemented on the IoT terminals, we conduct experiments on Android smart phones to act as terminals to interact with the DSSP. We perform the zero-knowledge proof process and calculate the parameters required for zero-knowledge proof with the time consumption recorded. The times to calculate the parameters for the first and second zero-knowledge proof are 254ms and 706ms respectively, and the results are averaged by 100-times simulations. According to the simulation result, the time consumption is acceptable with the consideration of the transmission delay.

(2) *Memory Consumption:* After analyzing the time cost, we also need to analyze the memory cost of zero-knowledge proof. Since the memory cost of performing single zero-knowledge proof is relatively small on both the DSSP side and the terminal side, we only test the memory cost while the DSSP performs multiple zero-knowledge proofs parallel. As shown in Fig. 5a, with the number of simultaneous verifications increasing, the memory consumption also increases. However, Fig. 5b describes that the average memory consumption decreases with the number of simultaneous verifications increasing. The reason is that at the begin of the zero-knowledge proof, there is a parameter initialization step, which will cost much memory. Then the following verification processes don't need the parameter initialization any more. Thus, the average memory consumption tends to be stable at nearly 13MB at last.

6.3 Additional Communication Overhead

In terms of communication, our proposed scheme adds two zero-knowledge proof processes to the original data sharing model. Thus, we quantitatively analyze the communication cost of the zero-knowledge proof processes. The elliptic curve of the bilinear pairing is fixed with a base field size of 512 bits and the order q is 160 bits. In the initialization step (shown in part C.1), the provider needs to obtain 1066 bytes of public parameters from the DSSP. At the beginning of awarding(shown in part C.2), a provider sends 148 bytes of hidden information to the DSSP to obtain anonymous credential. In this process, the provider also needs to send a 738-byte certificate and 128-byte signature used in the data transmission stage to verify the terminals' eligibility of reward. After the DSSP verifies the provider's eligibility for the award, it will send back anonymous credential to the provider with size 188 bytes. Then, the provider will use the anonymous credential to generate 20-byte R to receive the reward. In addition, the two zero-knowledge proofs will also cost 296-byte bandwidth and 820-byte bandwidth respectively. In conclusion, the additional communication cost totally 3404 bytes, which can be accepted.

7 Conclusion

With the volume of data increasing in the Internet of Things, a new business mode called crowd sensing has emerged. However, some security and privacy issues still exist that prevents the data owners from participating in the process. Especially, when a data provider claims rewards with its real identity for the shared data, the linkage between its real identity and the shared data will expose the participator's private information included in the shared data. To protect user's privacy in the scenario, this paper proposes a data sharing scheme with privacy protection based on anonymous certificate and zero-knowledge proof. We first use blockchain technology to improve the data sharing framework so that the behavior of all participants is supervised. Then, we propose an anonymous data sharing mechanism, in which participants can generate anonymous certificates for data sharing. With the anonymous certificate, the authenticity of messages transmitted in the network can be verified. In addition, in order to protect the privacy of participants while claiming rewards, two zero-knowledge proofs are utilized to ensure that the DSSP and malicious requestor will not link the data providers' real identities and their shared data content. At last, the simulation results demonstrate the efficiency and the feasibility of our scheme. In this paper, we only focus on the functionalities of the proposed scheme, without considering the scalability. In order to satisfy large volume of data sharing requests in the future, we will do more research on the scalability of the scheme.

References

[1] V. Mayer-Schönberger and K. Cukier, *Big Data: A Revolution That Will Transform How We Live, Work, and Think*. Boston, MA, USA: Houghton Mifflin, 2013.

[2] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.

[3] Y. Zhao, H. Haddadi, S. Skillman, S. Enshaeifar, and P. Barnaghi, "Privacy-preserving activity and health monitoring on databox," in *Proc. 3rd ACM Int. Workshop Edge Syst. Anal. Netw.*, 2020, pp. 49–54.

[4] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 577–590, Jul./Aug. 2016.

[5] M. Bayat, M. Doostari, and S. Rezaei, "A lightweight and efficient data sharing scheme for cloud computing," *Int. J. Electron. Informat. Eng.*, vol. 9, no. 2, pp. 115–131, 2018.

[6] A. Tchernykh *et al.*, "AC-RRNS: Anti-collusion secured data sharing scheme for cloud storage," *Int. J. Approx. Reasoning*, vol. 102, pp. 60–73, 2018.

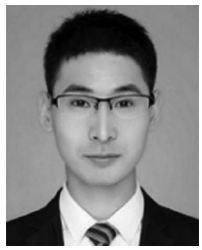
[7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized Bus. Rev.*, 2008, Art. no. 21260.

[8] H. Shafagh, L. Burkhalter, A. Hithnawi, and S. Duquennoy, "Towards blockchain-based auditable storage and sharing of IoT data," in *Proc. Cloud Comput. Secur. Workshop*, 2017, pp. 45–50.
[9] G. Srivastava, R. M. Parizi, A. Dehghantanha, and K.-K. R. Choo, "Data sharing and privacy for patient IoT devices using blockchain," in *Proc. Int. Conf. Smart City Informatization*, 2019, pp. 334–348.
[10] D. Hopwood, S. Bowe, T. Hornby, and N. Wilcox, "Zcash protocol specification," *GitHub*, San Francisco, CA, USA, 2016.
[11] S. Bajoudah, C. Dong, and P. Missier, "Toward a decentralized, trust-less marketplace for brokered IoT data trading using blockchain," in *Proc. IEEE Int. Conf. Blockchain*, 2019, pp. 339–346.
[12] Y. Huang, Y. Zeng, F. Ye, and Y. Yang, "Fair and protected profit sharing for data trading in pervasive edge computing environments," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 1718–1727.
[13] D. Hu, Y. Li, L. Pan, M. Li, and S. Zheng, "A blockchain-based trading system for big data," *Comput. Netw.*, vol. 191, 2021, Art. no. 107994.
[14] W. Chen, Y. Chen, X. Chen, and Z. Zheng, "Toward secure data sharing for the IoV: A quality-driven incentive mechanism with on-chain and off-chain guarantees," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 1625–1640, Mar. 2019.
[15] W. Gao, W. Yu, F. Liang, W. G. Hatcher, and C. Lu, "Privacy-preserving auction for big data trading using homomorphic encryption," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 2, pp. 776–791, Second Quarter 2018.
[16] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Blockchain empowered asynchronous federated learning for secure data sharing in Internet of Vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4298–4311, Apr. 2020.
[17] C. Niu, Z. Zheng, F. Wu, X. Gao, and G. Chen, "Trading data in good faith: Integrating truthfulness and privacy preservation in data markets," in *Proc. IEEE 33rd Int. Conf. Data Eng.*, 2017, pp. 223–226.
[18] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, 2019.
[19] H. Deng, Z. Qin, L. Sha, and H. Yin, "A flexible privacy-preserving data sharing scheme in cloud-assisted IoT," *IEEE Internet Things J.*, vol. 7, no. 12, pp. 11601–11611, Dec. 2020.
[20] J. Granjal, E. Monteiro, and J. S. Silva, "Security for the internet of things: A survey of existing protocols and open research issues," *IEEE Communications Surveys Tutorials*, vol. 17, no. 3, pp. 1294–1312, Third Quarter 2015.
[21] K. Fan *et al.*, "A secure and verifiable data sharing scheme based on blockchain in vehicular social networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 5826–5835, Jun. 2020.
[22] C. H. Liu, Q. Lin, and S. Wen, "Blockchain-enabled data collection and sharing for industrial IoT with deep reinforcement learning," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3516–3526, Jun. 2019.
[23] M. Z. Lee, A. M. Dunn, B. Waters, E. Witchel, and J. Katz, "AnonPass: Practical anonymous subscriptions," in *Proc. IEEE Symp. Secur. Privacy*, 2013, pp. 319–333.
[24] M. H. Au, W. Susilo, and Y. Mu, "Constant-size dynamic k-TAA," in *Proc. Int. Conf. Secur. Cryptogr. Netw.*, 2006, pp. 111–125.
[25] E. Androulaki *et al.*, "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proc. 13th EuroSys Conf.*, 2018, pp. 1–15.
[26] Y. Pu, C. Hu, S. Deng, and A. Alrawais, "R^2PEDS: A recoverable and revocable privacy-preserving edge data sharing scheme," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8077–8089, Sep. 2020.
[27] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 968–979, May 2020.
[28] J. Liu, Z. Zhang, X. Chen, and K. S. Kwak, "Certificateless remote anonymous authentication schemes for wireless body area networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 2, pp. 332–342, Feb. 2014.
[29] J. Shao, X. Lin, R. Lu, and C. Zuo, "A threshold anonymous authentication protocol for VANETs," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1711–1720, Mar. 2016.
[30] D. He, S. Zeadally, N. Kumar, and J.-H. Lee, "Anonymous authentication for wireless body area networks with provable security," *IEEE Syst. J.*, vol. 11, no. 4, pp. 2590–2601, Dec. 2017.

Chao Wang (Member, IEEE) received the PhD degree from the Beijing Institute of Technology, Beijing, China in 2015. He is currently an associate professor with the North China University of Technology, Beijing, China, with For more information, please visit wangchao.andy@gmail.com.

Shuo Wang is currently working toward the master's degree with the North China University of Technology. His research interests include the Internet of Vehicles and privacy in vehicle and blockchain.

Xiaoman Cheng is currently working toward the master's degree with the North China University of Technology. Her research interests include the blockchain and Internet of Vehicles.



Yunhua He (Member, IEEE) received the PhD degree in computer science from Xidian University, Xi'an, China, in 2016. He has been an associate professor with the North China University of Technology, China, since 2019. His research interests include security and privacy in cyber-physical systems.

Ke Xiao received the PhD degree in circuit and system from the Beijing University of Posts and Telecommunications, Beijing, China, in 2008. He has long been involved in the research and development and teaching work of wireless communications, Internet of Things, and embedded systems.

Shujia Fan received the master's degree from Boston University in 2018. She has been a research assistant with the Department of Computer Science and Technology of Tsinghua University since 2019. She has published several research articles in related international journals. Her current research interests include machine learning, blockchain, and digital finance.