Blockchain-Based Privacy-Preserving and Rewarding Private Data Sharing for IoT

Tian Li, Huaqun Wang, Debiao He, *Member, IEEE*, and Jia Yu, *Member, IEEE*

***Abstract*—The Internet of Things (IoT) devices possessed by individuals produce massive amounts of data. The private data onto specific IoT devices can be combined with intelligent platform to provide help for future research and prediction. As an important digital asset, individuals can sell private data to get rewards. Problems, such as privacy, security, and access control prevent individuals from sharing their private data. The blockchain technology is widely used to build an anonymous trading system. In this article, we construct a blockchain-based privacy-preserving and rewarding private data-sharing scheme (BPRPDS) for IoT. A privacy issue worth considering is that the malicious cloud server may establish a behavior profile database of data users (DUs). In the case of anonymity, the transactions of private data sharing are easy to cause disputes. When anonymous DUs are framed, it is hard to protect their rights. With the help of the deniable ring signature and Monero, we realize the behavior profile building prevention and nonframeability of BPRPDS. At the same time, we utilize the licensing technology executed by smart contracts to ensure flexible access control of multisharing. The proposed BPRPDS is provably secure. Performance analysis and experimental results show that BPRPDS is efficient and practical.**

***Index Terms*—Anonymity, blockchain, data sharing, ring signature.**

I. Introduction

INTERNET of Things (IoT) technology is closely related to our life. It has been widely used in a series of fields, including smart home [1], health care [2], intelligent transportation [3], and industrial manufacturing [4]. With the development of 5G network, this growth trend is more and more obvious. According to Gartner's forecast, there will be 41.6 billion IoT smart devices in the world by 2025. At the same time, a faster 5G network means that the massive data collected from IoT devices will be analyzed and managed more efficiently [5]. The traditional IoT systems rely on data centers (e.g., cloud servers) to store and process data. This assumption of trust in cloud servers makes outsourcing data face serious security and privacy risks, especially for individual IoT users.

As the digital assets of individuals, private data onto IoT devices can bring benefits to individuals. Take the healthcare scenario of IoT for example. The application of the IoT technology in the healthcare scene begins with the remote monitoring of patients. Wearable medical devices are embedded with wireless sensors to collect and transmit a series of real-time body parameters from patients [6]. With the popularity of wearable devices, ordinary people can also be encouraged to share these data with medical institutions for basic research. Individuals who contribute valuable IoT data should be rewarded.

The blockchain technology can assist IoT system in terms of privacy, traceability, and interoperability [7]. Blockchain is a distributed ledger with a chain structure. It comes from the bitcoin cash system. It uses the consensus mechanism to make the data on the blockchain unforgeable, and utilizes the pseudonym mechanism to achieve the anonymity of users [8]. The smart contract technology accelerates the application of blockchain. A smart contract defines the agreement on the two sides of the communication, and automatically executes the transaction in the form of executable code according to the agreement [9]. However, the input data and output data onto the smart contract are transparent in the blockchain. If these data are secret values, the existing blockchain systems will easily expose the privacy of users. The trusted execution environment (TEE) assists blockchain nodes to perform private operations of smart contracts by isolating the external environment, which can ensure the integrity and privacy of data onto smart contracts [10]. For example, an enclave container in the Intel SGX technology can load key codes and process sensitive data [11]. TEE can help blockchain systems improve security and performance, including key management [12], data sharing [13], and privacy protection [14]. The blockchain technology is also widely used in digital currency [15]–[17], energy trading [18], and vehicle network [19], [20].

A. Motivation

We consider a specific healthcare scenario of IoT: an individual sells his IoT data (e.g., ECG data, blood glucose data,

Manuscript received 9 March 2021; revised 24 June 2021 and 10 December 2021; accepted 24 January 2022. Date of publication 31 January 2022; date of current version 8 August 2022. This work was supported by the National Natural Science Foundation of China under Grant 61872192, Grant 61972294, Grant 61932016, and Grant U21A20466. *(Corresponding author: Huaqun Wang.)*

Tian Li is with the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail: litian_99@163.com).

Huaqun Wang is with the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China, and also with the Department of Computer Science and Engineering, Shaoxing University, Shaoxing 312000, China (e-mail: wanghuaqun@aliyun.com).

Debiao He is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China (e-mail: hedebiao@163.com).

Jia Yu is with the College of Computer Science and Technology, Qingdao University, Qingdao 266071, China (e-mail: qduyujia@gmail.com).

Digital Object Identifier 10.1109/JIOT.2022.3147925

and blood pressure data) to medical research institutions. Due to the openness of the IoT system, the security, and privacy of private data sharing is bound to be threatened by adversaries. In the actual IoT environment, an attractive private data-sharing system should meet the following goals.

1) *Anonymity:* In the process of data sharing, the most individuals are unwilling to reveal their true identity. That is because the sensing data contains a lot of body information about the data owner (DO), and researchers may infer potential diseases. If the anonymity of DOs cannot be protected, it may lead to unwillingness of individual users to share data with medical institutions. Similarly, medical institutions maintain anonymity because of their secret researches.
2) *Data Authenticity:* If the shared IoT data is forged by the adversary, it will mislead the research institutions. Especially in the case of anonymity, we need a trusted method to verify the authenticity of the shared data.
3) *Rewarding:* To encourage private data sharing, individuals who contribute sensing data should be rewarded.
4) *Behavior Privacy of Data User (DU):* The behavior privacy of the medical research institution (i.e., DU) also needs to be guaranteed. That is because malicious cloud servers may build the behavior profile database of DU based on its access records. Although the DU is anonymous, the malicious CS may record the access frequency and time of its public key.
5) *Access Control:* Individuals have no special security platform similar to enterprises. In the process of getting rewards through multisharing, they need a trusted carrier, which can easily set sales rules and access policies.
6) *Nonframeability:* When an anonymous DU is maliciously framed, there can be a denial mechanism to protect its rights. The framed DU can submit the proof to the trusted regulatory authority, but it does not affect other security attributes.

The above-mentioned aspects inspire our research work. As far as we know, there is little literature on privacy-preserving and rewarding private data sharing, and they do not consider the behavior privacy and nonframeability of DUs. The general ledger technology and consensus mechanism guarantee that the data onto the blockchain is unforgeable. Therefore, DUs can verify the authenticity of shared data according to the metadata in the blockchain. Blockchain can establish peer-to-peer transactions between DOs and DUs. The blockchain technology can eliminate the profit of middlemen and help DOs get the most reward. In the blockchain, DOs are free to price and set the access rights (e.g., time limited access) of DUs. At the same time, the smart contract can act as the licence server to issue licences for DUs. There are three types of blockchain: 1) public blockchain (e.g., Bitcoin); 2) consortium blockchain (e.g., Hyperledger Fabric); and 3) private blockchain. Public blockchain has low scalability due to its high dependence on the consensus mechanism. For example, the throughput of Bitcoin is about 7 transactions per second (TPS) [21]. Presently, the throughput of Hyperledger Fabric can reach 20 000 TPS [22]. A private blockchain is a kind of private network used for internal management [23]. The nodes on the private blockchain are completely controlled by an organization, and their operations cannot be hidden. Compare with the other two types, the optional privacy ability and fast transaction ability of consortium blockchain attracts our attention. Therefore, the privacy-preserving and rewarding private data-sharing scheme based on the consortium blockchain is a subject worth studying.

B. Related Work

Liang *et al.* [24] used the proxy re-encryption technology to realize multisharing controls of ciphertext data for big data storage. Liang *et al.* [24] does not exposed data knowledge and identity information of sender/receiver. Huang *et al.* [25] constructed an anonymous data-sharing scheme with forward security by using ring signature. However, this scheme does not discuss the access control of users. According to the realistic scenario of enterprise acquisition, Wang *et al.* [26] proposed a secure and effective provable data possession scheme to ensure the integrity of data in the process of data transfer. Fan *et al.* [27] proposed a data sharing and privacy computing scheme based on ciphertext policy attribute encryption, which has effective user attribute revocation mechanism. The disadvantage of this scheme is the heavy communication overhead caused by fine-grained access control, and it can not resist the attack of malicious revocation of legitimate users in the cloud. In order to eliminate the single point performance bottleneck in data sharing, Xue *et al.* [28] proposed an auditable multiauthority access control scheme.

The data management system assisted by blockchain is reliable. That is because the data on-chain cannot be forged to prevent data from being abused. Xu *et al.* [29] proposed a mobile network identity management and authentication scheme based on blockchain, which ensures that sensitive data is controlled by users themselves. Luo *et al.* [30] designed a secure data aggregation scheme with homomorphic encryption on the blockchain, which uses the particle swarm optimization algorithm and smart contract for automatic power dispatching. In recent years, blockchain-assisted data sharing in various fields has been widely studied. Sun *et al.* [31] designed a tamper-resistant data-sharing scheme for vehicular network with the help of the blockchain technology. Huang *et al.* [32] proposed a privacy-preserving medical data-sharing scheme based on blockchain by using technologies of the proxy re-encryption and zero knowledge proof. For Industrial IoT, Lu *et al.* [33] transformed the data-sharing problem of privacy protection into machine learning problem by combining blockchain and federated learning. Xu *et al.* [34] constructed a large-scale health data privacy-preserving chain (named healthchain) to encrypt health data for fine-grained access control. So far, there are few incentive private data-sharing schemes for privacy-preserving. Wang *et al.* [35] developed a TEE-based smart contract execution mechanism on the blockchain, which is applied to the incentive private data-sharing scheme for fine-grained access control in the smart grid.

C. Our Contribution

Based on the blockchain, we have implemented a more secure and practical blockchain-based privacy-preserving and rewarding private data-sharing scheme (BPRPDS) scheme. Our contributions are given as follows.

1) We propose a privacy-preserving incentive private data-sharing scheme on the blockchain for the first time. The DO publishes the private data anonymously and gains the reward anonymously in the blockchain; DUs buy and obtain licences anonymously through smart contracts, where the licensing technology ensures access control for multisharing. Then, we use the Monero technology to realize the untraceability and unlinkability of DUs (even their public keys) in the process of obtaining private data. Thus, no one can build the behavior profile database of DUs.
2) We introduce the security attribute of nonframeability into the anonymous incentive data-sharing scheme for the first time. With the help of a deniable ring signature, honest DUs can deny the frame up without revealing their true identity.
3) We give the system model and security model of BPRPDS. Under the random oracle model, we give the formal security proof. Through performance analysis and experimental simulation, BPRPDS is effective and attractive.

D. Paper Organization

The remainder of this article is organized as follows. Section II gives some preliminaries. Section III shows the formal system model and security definition of our BPRPDS scheme. Section IV presents the detailed BPRPDS scheme. Section V evaluates the correctness, security, and performance of BPRPDS. Finally, Section VI summarizes the main findings and future work.

II. Preliminaries

In this section, we review some cryptography knowledge, including bilinear mapping, computational hard problems, deniable ring signature, Monero, and digital rights management (DRM).

A. Bilinear Pairings and Computational Hard Problems

Definition 1 (Bilinear Mapping): Let $(\mathbb{G}_1, +)$ and $(\mathbb{G}_2, \cdot)$ be cyclic groups of prime order q, and q is a prime number. Then the map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is called computational bilinear mapping if e has the following properties.

1) *Bilinearity:* $\forall\ P, P_1, P_2 \in \mathbb{G}_1$, and $a, b \in Z_q^*$
$$e(P, P_1 + P_2) = e(P, P_1)e(P, P_2)$$
$$e(aP_1, bP_2) = e(P_1, P_2)^{ab}.$$
2) *Nondegeneracy:* $\exists R_1, R_2 \in \mathbb{G}_1$ such that $e(R_1, R_2) \neq 1_{\mathbb{G}_2}$
3) *Computability:* $\forall R_1, R_2 \in \mathbb{G}_1$, there is an efficient time algorithm to compute $e(R_1, R_2)$.

Bilinear mapping satisfying the conditions described above can be constructed by using changed Weil pairings or Tate pairings on elliptic curves [36]. Difficult problems based on bilinear pairings are defined as follows.

Definition 2 (Discrete Logarithm Problem (DLP)]: Given $R_1, R_2 \in \mathbb{G}_1$, find an integer $c \in Z_q^*$ that satisfies $R_1 = cR_2$.

Definition 3 [Computation Diffie–Hellman Problem (CDHP)]: Let $P \in \mathbb{G}_1$, $a_1, a_2 \in Z_q^*$, given P, a_1P, a_2P, compute a_1a_2P.

B. Deniable Ring Signature and Monero

Komano *et al.* [37] proposed the deniable ring signature which is used to solve the problem that the nonactual signer is framed because of the unconditional anonymity in the ring signature. In [37], anonymity is conditional. In other words, the signer can confirm that the signature is related to himself/herself through a confirmation algorithm. The nonsigner can refute that the signature is related to himself/herself through a disavowal algorithm.

Monero is one of the digital currencies with privacy characteristics based on blockchain [15], [16]. Monero guarantees the unlinkability of the receiver by the stealth address, and guarantees the untraceability of the sender by the ring signature. Thus, Monero ensures the privacy of transactions.

C. DRM

DRM is a series of operations that middlemen (e.g., platform vendors or copyright providers) use the licensing technology to control the digital content of consumers, such as access control, sharing, and replication. Win *et al.* [38] proposed a DRM scheme based on the blind decryption and anonymous cash without relying on the third party. Win *et al.* [38] can revoked malicious users without violating their privacy. The disadvantage of [38] is that the user's behavior can be observed. Zhang and Zhao [39] proposed a DRM mechanism using the smart contract to ensure automatic licences issuance and free pricing rules. More and more attention has been paid to the licensing technology in DRM without encryption due to the popularity of various intelligent terminals. Presently, many privacy-preserving data sharing and content distribution schemes adopt the DRM technology [40]–[42].

III. System Definition and Security Requirements

In this section, we first demonstrate the system definition of BPRPDS. Then, we give the security model of BPRPDS based on the security requirements.

A. System Model

As shown in Fig. 1, we present the general architecture of private IoT data sharing in the healthcare scenario mentioned in Section I-A. There are five entities in BPRPDS: 1) DO; 2) DU; 3) management center (MC); 4) CS; and 5) blockchain. Their roles in the system are described as follows.

1) *DO:* The individual with valuable IoT data, who is also the direct beneficiary of private data sharing.
2) *DU:* The entity willing to pay for the private data onto DO. In the process of buying or data acquisition, it wants to keep its behavior hidden.
3) *MC:* A trusted entity responsible for verifying the licence message submitted by the DU. At the same

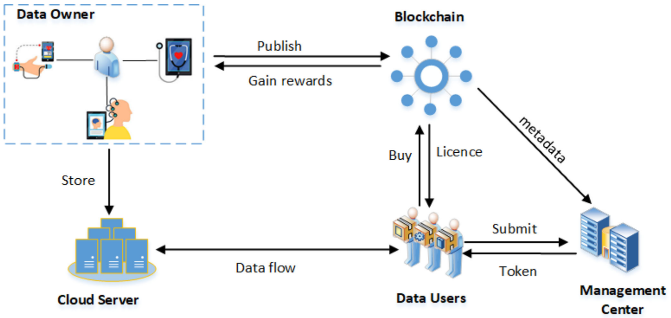


Fig. 1. Overview architecture of BPRPDS.

time, it is responsible for generating the stealth address and access token for the successfully authenticated DU. When a DU is framed, it also acts as an arbiter to verify the DU's disavowal algorithm. In real-world IoT applications, the MC may be an audit authority or regulatory authority.

4) *CS:* An entity with rich storage and computing capabilities that stores the ciphertext of shared data.
5) *Blockchain:* A nontampered entity that protects the privacy of transactions. On the blockchain, the DO and DU can maintain anonymity in the transaction of private data sharing. The DO sets sales rules and access policies on the blockchain. Smart contracts provide licensing services for multisharing. For the privacy and scalability of BPRPDS, we adopt the consortium blockchain.

B. Security Definitions

Based on the aforementioned healthcare example of private IoT data sharing, we refine the following security attributes.

1) *Anonymity:* DO needs to keep anonymity when publishing his/her private IoT data and getting rewards. In the process of payment, licence enforcement, and data acquisition, DUs' identity cannot be recognized by anyone. When an honest DU is framed, it can submit the proof to the MC through its public key. However, this proof does not reveal the real identity of the DU.
2) *Unforgeability:* Take any a group of public keys and message as input, no one can forge a signature to pass the MC's verification.
3) *Nonframeability:* When a DU is framed, it can prove to the MC that it is honest through the disavowal algorithm.
4) *Profile Building Prevention of DUs:* During the phases of buying, licence enforcement, and data acquisition, no observer can establish the behavior profile database of DUs.

For the convenience of proof, we introduce some oracles used in the formal security definitions.

1) *Random Oracle ($\mathcal{O}_{\mathcal{R}}$):* Taking a number as input, a random value is returned.
2) *Corruption Oracle ($\mathcal{O}_{\mathcal{CO}}$):* Taking a public key X_i as input, the corresponding private key x_i is returned.
3) *Signature Oracle ($\mathcal{O}_{\mathcal{S}}$):* Taking a group of public keys Π and message m as input, a valid signature σ is returned.
4) *Disavowal Oracle ($\mathcal{O}_{\mathcal{D}}$):* Taking a message m, a signature σ as input, the oracle uses the private key x_i to prove σ is not associated with the public key X_i. The challenger does not output the query result of the target signature.

Then, we give the formal security definitions of BPRPDS.

Definition 4 (Anonymity): For any probability polynomial time (PPT) adversary $\mathcal{A}$ even with unlimited computing power, our BPRPDS scheme satisfies the anonymity of DO and DU if the following scenarios hold.

1) In phases anonymous publishing and anonymous buying, $\mathcal{A}$ can guess the identity of DO and DU with negligible probability.
2) In the phase anonymous licence enforcement, anonymity of signers in ring signature is defined in the game between the adversary $\mathcal{A}$ and the challenger $\mathcal{C}$. $\mathcal{C}$ generates system parameters for $\mathcal{A}$ and $\mathcal{A}$ is allowed to access the oracles $\mathcal{O}_{\mathcal{R}}$ and $\mathcal{O}_{\mathcal{CO}}$. For any group of n public keys, any message m and any valid signature, the probability of $\mathcal{A}$ can guess the identity of real signer is not more than $(1/n)$.
3) In the phase anonymous data acquisition, $\mathcal{A}$ can guess the DU's identity with negligible probability.
4) In the phase prevention of frame up, the probability of $\mathcal{A}$ obtaining the identity from the proof submitted by the framed DU is negligible.

Definition 5 (Unforgeability): Unforgeability of BPRPDS is defined in the game between the adversary $\mathcal{A}$ and the challenger $\mathcal{C}$. $\mathcal{C}$ generates system parameters for $\mathcal{A}$ and $\mathcal{A}$ is allowed to access the oracles $\mathcal{O}_{\mathcal{R}}$, $\mathcal{O}_{\mathcal{CO}}$, and $\mathcal{O}_{\mathcal{S}}$. If the probability that $\mathcal{A}$ can win the following game is negligible, our BPRPDS scheme satisfies unforgeability.

1) *Setup:* $\mathcal{C}$ runs *Setup* to provide system parameters for $\mathcal{A}$. All the public keys and system parameters are public to $\mathcal{A}$.
2) *Hash Queries:* $\mathcal{A}$ submits hash queries to $\mathcal{C}$ adaptively, and $\mathcal{C}$ returns the response of $\mathcal{O}_{\mathcal{R}}$. If it is a fresh query, $\mathcal{C}$ stores the response in a hash table Q_H (initialized to empty). Otherwise, $\mathcal{C}$ returns the recorded result from Q_H.
3) *Signature Query:* $\mathcal{A}$ chooses a set Π^* of n^* public keys, and the message m^*. Then, $\mathcal{A}$ submits the tuple (Π^*, m^*) to make a query, a valid signature σ^* is returned from $\mathcal{O}_{\mathcal{S}}$.
4) *Forge:* $\mathcal{A}$ forges a valid signature $\hat{\sigma}^*$ on $(\hat{\Pi}^*, \hat{m}^*)$, where $(\hat{\Pi}^*, \hat{m}^*)$ are not queried to $\mathcal{O}_{\mathcal{S}}$. We say $\mathcal{A}$ wins the game if

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{for}}(\lambda) = \Pr\left[\begin{matrix}\mathrm{Verfiy}(\hat{\Pi}^*, \hat{m}^*, \\ \hat{\sigma}^*) = \mathrm{valid}\end{matrix}\right] \geq \varepsilon_A(\lambda)$$

where $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{for}}$ is the advantage function of $\mathcal{A}$ winning the game in the forge attack and $\varepsilon_A(\lambda)$ is the trivial probability on the security parameter λ.

Definition 6 (Nonframeability): The adversary $\mathcal{A}$ is given the public key set $\Pi_{pk} = \{X_1, X_2, \ldots, X_n, \ldots, X_{n'}\}$ in the whole system, and $\mathcal{A}$ is allowed to access the oracles $\mathcal{O}_{\mathcal{R}}$, $\mathcal{O}_{\mathcal{CO}}$, $\mathcal{O}_{\mathcal{S}}$, and $\mathcal{O}_{\mathcal{D}}$ to make adaptive queries. $\mathcal{A}$ outputs a valid signature σ_k on (Π_{pk}, m), where the picked public key

TABLE I
NOTATIONS AND DESCRIPTIONS IN OUR BPRPDS SCHEME

Notations	Description
$\mathbb{G}_1$	A cyclic additive group with prime number q
$\mathbb{G}_2$	A cyclic multiplication group with prime number q
e	A bilinear pairing map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$
(d_o, D_o)	The private/public key pair of DO on the blockchain
(x_l, X_l)	The private/public key pair of the lth DU on the blockchain
(z, Z)	The private/public key pair of MC on the blockchain
H,H_1	Two different cryptographic hash functions
P,Q	Two different generators of $\mathbb{G}_1$
(Sig, Ver)	The signature/verification algorithm on the blockchain
$(\varepsilon_{pk}, \mathcal{D}_{sk})$	The public key encryption/ decryption algorithm
(E_{sym}, D_{sym})	The symmetric key encryption/decryption algorithm
Π	The public key set in the ring signature
$\|\Pi\|$	The cardinality of Π

$X_k \in \Pi_{pk}$ is uncorrupted. The challenger $\mathcal{C}$ carries out the disavowal algorithm using the public key X_k. We say $\mathcal{A}$ wins the game if

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{non}-f}(\lambda) = Pr\big[\mathrm{Disavowal}\big(\Pi_{pk}, m, \sigma_j\big) = \mathrm{fail}\big] \geq \varepsilon_A(\lambda)$$

where $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{non}-f}$ is the advantage function of $\mathcal{A}$ winning the game in the frame attack and $\varepsilon_A(\lambda)$ is the trivial probability on the security parameter λ.

Definition 7 (Profile Building Prevention): In our BPRPDS scheme, any PPT adversary $\mathcal{A}$ can build the behavior profile database of DUs (even their public keys) with negligible probability.

In order to understand the notations in this article, we present them and their descriptions in Table I.

IV. Our Concrete BPRPDS Scheme

In this section, we propose a specific scheme. To further illustrate our BPRPDS scheme, we refine the interactions between entities as shown in Fig. 2. As shown in Fig. 2, the interactions among these entities are as follows.

1) System initialization.
2) The DO stores the ciphertext of private data on the CS.
3) The DO sets the sale rules and access control policies of private data sharing on the blockchain. Then, the DO publishes them on the blockchain.
4) The DU buys the private data of DO and gets the licence after successful payment. At the same time, the DO gains the rewards.
5) The DU generates the ring signature for the licence message by mixing with other public keys and submits it to the MC. Through the secret materials of the signature, the MC generates the stealth address and access token for the DU.
6) The DU submits the access token to CS and obtains data through the stealth address.
7) When the DU is framed, it can prove to the MC that it is not the real signer through the disavowal algorithm.

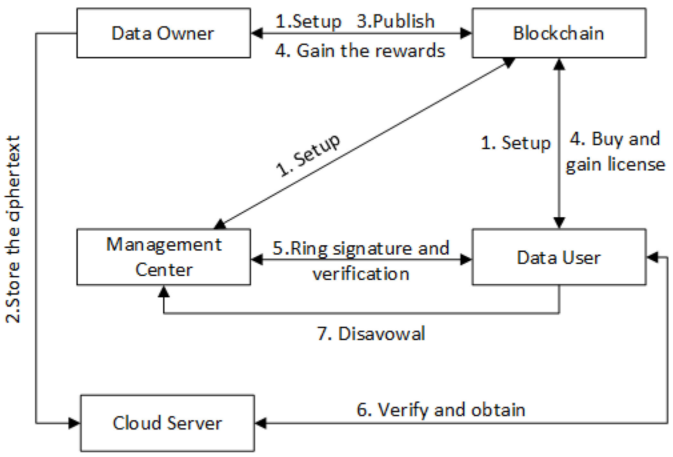


Fig. 2. Interactions among entities in BPRPDS.

A. Setup

The security parameter of the system is defined as λ. Let $\mathbb{G}_1$ be a cyclic additive group with the prime number q and $\mathbb{G}_2$ be a cyclic multiplication group with the prime number q. e is denoted as a mapping $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ $\forall P, Q \in \mathbb{G}_1$. H and H_1 defined below are two different secure cryptographic hash functions

$$H : \{0, 1\}^* \to Z_q^*$$
$$H_1 : \mathbb{G}_1 \to Z_q^*.$$

1) The DO picks a random number $d_o \in Z_q^*$, and computes $D_o = d_o P$ as his/her public key. Thus, the private/public key pair of DO is (d_o, D_o). The public key D_o is associated with the DO's address on the blockchain.
2) Denote the lth DU as DU_l. DU_l picks a random number $x_l \in Z_q^*$, and computes its public key $X_l = x_l P$, where $l \in [1, n']$. Thus, the private/public key pair of DU_l is (x_l, X_l). The public key X_l is associated with the DU_l's address on the blockchain.
3) The MC picks a random number $z \in Z_q^*$, and computes its public key $Z = zP$. Thus, the private/public key pair of MC is (z, Z).

Therefore, the system parameters $(\mathbb{G}_1, \mathbb{G}_2, P, Q, \ e, D_o, X_l, Z)$ are public and stored on the blockchain. The private keys of all entities are confidential. In addition, we define several secure algorithms on the blockchain: the signature/verification algorithm (Sig, Ver), the symmetric key encryption/decryption algorithm $(E_{\mathrm{sym}}, D_{\mathrm{sym}})$, and the public key encryption/decryption algorithm $(\varepsilon_{pk}, \mathcal{D}_{sk})$.

B. Anonymous Publishing

Denote the private data onto DO as M. The DO performs the following.

1) Encrypt M with a symmetric key K as $C_M = E_K(M)$ using $E_{\text{sym}}(\cdot)$.
2) Compute $M^* = H(M)$ and upload it to the blockchain.
3) Store (C_M, M^*) on the CS.
4) Publish the price and sale rules on the blockchain. At the same time, some sensitive parameters (e.g., K) are written into the smart contract executed in TEE, and these parameters are saved in the special container of TEE (e.g., enclave of Intel SGX).

In order to prevent DO from publishing worthless IoT data, the DO deposits transaction margin on the MC.

C. Anonymous Buying

In this phase, we use the smart contract to make the anonymous payment between the DO and the DU. After successful payment, the smart contract issues a licence to the DU.

When a DU is interested in the private data onto DO and accepts the sale rules, it initiates a purchase request. Denote the buying DU as DU$_l$. Upon receiving the buying request from DU$_l$, the smart contract carries out in TEE as follows.

1) Check the balance on the account address of DU$_l$. If sufficient, goto the next step; otherwise, terminate the transaction.
2) Generate the licence Lice for DU$_l$, where

$$\text{Lice} = \left(\text{seq}_l, M^*, X_l, K, CR, t_s\right).$$

   a) seq$_l$ is the unique identification of Lice.
   b) CR includes usage rights (e.g., store, browse, or play) and the expiration time of Lice.
   c) t_s is the time stamp of Lice.
3) Compute $m = H(\text{Lice})$. Then, send the tuple (M^*, m, CR) to the MC.
4) Encrypt Lice with DU$_l$'s public key X_l as $CL = \varepsilon_{X_l}(\text{Lice})$. Then, send (m, CL) to DU$_l$.

D. Anonymous Licence Enforcement

In this phase, DU$_l$ submits the licence message m to the MC. In order to anonymous authentication, DU$_l$ mixes with other $(n-1)$ public keys in the system to form a group $\Pi = \{X_1, X_2, \ldots, X_n\}$, where $|\Pi| = n$.

Signing: DU$_l$ performs the following.

1) Pick two random values ϕ and ψ from Z_q^*, and then compute $\Phi = \phi P$, $\Psi = \psi P$.
2) Compute $T = x_l Q$, $\mu_0 = H(\Pi, m, T, 0)$, $\mu_1 = H(\Pi, m, T, 1)$, $\Theta = (\mu_0 + \mu_1)Q$.
3) Compute $h_i = H(\Pi, m, U_i, \Phi, \Psi)$, where U_i is arbitrarily selected from $\mathbb{G}_1$, $i = 1, 2, \ldots, n$ and $i \neq l$.
4) $\forall r \leftarrow Z_q^*$, compute $U_l = rX_l - \sum_{i \neq l}^{n}(U_i + h_i X_i)$, $h_l = H(\Pi, m, U_l, \Phi, \Psi)$, $V = x_l(h_l + r)\Theta$.
5) Generate a ring signature $\sigma = (T, U_1, \ldots, U_n, V)$ on the message m and then send (m, σ, Φ, Ψ) to the MC.

E. Anonymous Token Acquisition

In the phase, the MC verifies the DU$_l$'s licence message and issues an access token to a stealth address for the successfully verified DU$_l$.

Verification: The MC verifies the signature of DU$_l$ as follows.

1) Compute $\mu_0 = H(\Pi, m, T, 0)$, $\mu_1 = H(\Pi, m, T, 1)$, $\Theta = (\mu_0 + \mu_1)Q$.
2) Compute $h_i = H(\Pi, m, U_i, \Phi, \Psi)$, for $i = 1, 2, \ldots, n$.
3) Verify whether the following equation holds:

$$e(P, V) = e\left(\sum_{i=1}^{n}(U_i + h_i X_i), \Theta\right).$$

When the exposed m and σ are valid, the MC generates the stealth address and access token for DU$_l$ as follows.

1) Select a random number $\delta \in Z_q^*$, and then compute a stealth address $\hat{X}_l = H_1(\delta\Phi)P + \Psi$ and $Y = \delta P$ for DU$_l$. Then, store Y to the blockchain.
2) Generate an access token $AT_l = (M^*, CR)$ and its signature $\text{Sig}_z(AT_l)$.
3) Send $(AT_l, \text{Sig}_z(AT_l))$ to $\hat{X}_l$.

Unlike the classical deniable ring signature scheme [37], [43], our scheme does not include the confirmation algorithm. That is because in the practical application, we do not need to consider the scenario where the signer actively proves that it is the real signer. Besides, the stealth address generated by MC depends on some secret materials in the signature that only the signer itself knows.

F. Anonymous Data Acquisition

In the phase, DU$_l$ interacts with the CS to obtain data. The specific process is as follows.

1) DU$_l$ gets Y from the blockchain, and computes $\hat{x}_l = H_1(\phi Y) + \psi$ which satisfies $\hat{x}_l P = (H_1(\phi Y) + \psi)P = H_1(\delta\Phi)P + \Psi = \hat{X}_l$. Thus, DU$_l$ obtains $(AT_l, \text{Sig}_z(AT_l))$ because it knows the private key $\hat{x}_l$ of the address $\hat{X}_l$.
2) DU$_l$ submits the token-signature pair $(AT_l, \text{Sig}_z(AT_l))$ to the CS through $\hat{X}_l$.
3) The CS verifies the signature $\text{Sig}_z(AT_l)$ by using the MC's public key Z and the verification algorithm *Ver*. If valid, the CS resolves M^* and CR in AT_l; otherwise, it rejects the service.
4) DU$_l$ runs the secure public key decryption algorithm $\varepsilon_{x_l}(CL)$ with its private key x_l to get the symmetric key K.
5) DU$_l$ obtains M through the secure symmetric key decryption algorithm $D_K(C_M)$ using K. Then, DU$_l$ verifies whether the equation $M^* = H(M)$ holds. If it holds, DU$_l$ accepts M; otherwise, it rejects M.

The token can be kept in the CS until the expiration time is included in CR. Once expired, the token will be revoked.

G. Prevention of Frame Up

There exists the situation that DUs are maliciously framed, such as license messages are stolen by attackers or leaked by smart contracts. In the case of anonymity, it is difficult for

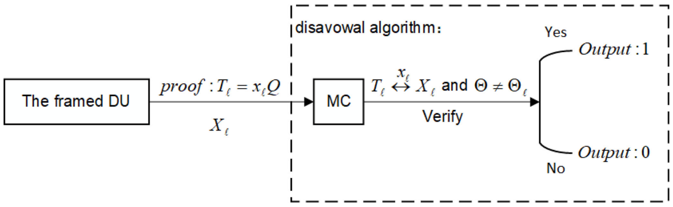

disavowal algorithm:
The framed DU
$proof: T_\ell = x_\ell Q$
X_ℓ
MC
$T_\ell \overset{x_\ell}{\leftrightarrow} X_\ell$ and $\Theta \neq \Theta_\ell$
Verify
Yes
$Output: 1$
No
$Output: 0$

Fig. 3. Interactions between the framed DU and the MC.

DUs to appeal their rights. We provide a disavowal algorithm to prevent honest DUs from being framed. When a DU is not the real signer on the licence message m, it executes the disavowal algorithm to proof its honesty. As shown in Fig. 3, the interactions between the framed DU and the MC are as follows.

Disavowal: About the signature σ on (Π, m), the nonsigner $\ell \in \Pi$ generates the proof $T_\ell = x_\ell Q$ and send it to the MC.

The MC performs the following verification.

1) $\forall d \leftarrow Z_q^*$, verify the following equation:

$$e(T_\ell, dP) = e(dQ, X_\ell).$$

(Correctness): $e(T_\ell, dP) = e(x_\ell Q, dP) = e(dQ, X_\ell)$. If the equation holds, it shows that T_ℓ and the public key X_ℓ are consistent in the private key x_ℓ, goto the next step. Otherwise, it outputs $\perp$.

2) Compute $\mu_0 = H(\Pi, m, T_\ell, 0)$, $\mu_1 = H(\Pi, m, T_\ell, 1)$, and $\Theta_\ell = (\mu_0 + \mu_1)Q$. If $\Theta \neq \Theta_\ell$ and T_ℓ is valid, the MC accepts the proof and outputs 1; otherwise, it outputs 0.

V. Analysis of Our Scheme

A. Correctness and Security Analysis

In this section, we give the proof of correctness. With the help of mathematical difficulties and random oracles, our scheme is provably secure. Some well-known mathematical difficulties are the basis of security proof in public-key cryptosystems. DLP and CDHP mentioned in Section II are among them.

Theorem 1 (Correctness): If the DO, DU, and MC execute the procedure honestly, the DU can pass the MC's verification successfully.

Proof: Correctness for ring signature verification

$$\begin{aligned} e\left(\sum_{i=1}^{n}(U_i + h_i X_i), \Theta\right) &= e\left(U_l + h_l X_l + \sum_{i \neq l}^{n}(U_i + h_i X_i), \Theta\right) \\ &= e(rX_l + h_l X_l, \Theta) \\ &= e((r + h_l)x_l P, \Theta) \\ &= e(P, V). \end{aligned}$$

■

Theorem 2 (Anonymity): Our scheme satisfies the anonymity of DO and DU for any PPT adversary $\mathcal{A}$.

Proof: We prove the anonymity for DU in the following five phases.

1) In the phase anonymous buying, the payment address associated with the DU's public key is a random point on $\mathbb{G}_1$.
2) In the phase anonymous licence enforcement, there are n ring members in the group, the signature generated by DU is $\sigma = (T, U_1, \ldots, U_n, V)$. T is a random point on $\mathbb{G}_1$, $(U_1, U_2, \ldots, U_{l-1}, U_{l+1}, \ldots, U_n)$ are n-1 random picked points on $\mathbb{G}_1$. $U_l = rX_l - \sum_{i \neq l}^{n}(U_i + h_i X_i)$ and $V = (h_l + r)\Theta$ are also random points, because r is randomly picked by DU, h_i is output by the hash function H and Θ is a random point on $\mathbb{G}_1$. Thus, the signature does not leak any information about the identity of DU.
3) In phase anonymous token acquisition, the DU is anonymous based on the fact that Φ and Ψ are randomly selected on $\mathbb{G}_1$.
4) In the phase anonymous data acquisition, the DU gets the data through the stealth address on the blockchain. The stealth address has nothing to do with the identity of the DU.
5) In the phase prevention of frame up, the framed DU submits T to the MC, where T is a random point on $\mathbb{G}_1$.

Thus, $\mathcal{A}$ can guess the identity of DU no better than $(1/n)$ based on n ring members. The anonymity of blockchain ensures that the DO is anonymous in the process of data publishing and buying. The theorem is proved. ■

Theorem 3 (Unforgeability): For any PPT adversary $\mathcal{A}$ even if it has unlimited computing power, our scheme satisfies the unforgeability if the CDHP assumption holds.

Proof: The interactions between the adversary $\mathcal{A}$ and the challenger $\mathcal{C}$ are as follows.

SetUp: $\mathcal{C}$ forwards the public parameters of the system to $\mathcal{A}$. $\mathcal{A}$ chooses the public keys group $\Pi = \{X_1, X_2, \ldots, X_n\}$, and there exists an index $j \in \{1, 2, \ldots, n\}$ that satisfies $X_j = aP$. $\mathcal{C}$ maintains a table Q_H initialized as empty.

Query on $\mathcal{O}_{\mathcal{R}}$: $\mathcal{A}$ requests a hash query for b. If this is a fresh query, $\mathcal{C}$ randomly picks $\xi \leftarrow Z_q^*$. Then, $\mathcal{C}$ adds (ξ, b) to Q_H and forwards them to $\mathcal{A}$; otherwise, $\mathcal{C}$ returns the recorded (ξ, b) to $\mathcal{A}$.

Query on $\mathcal{O}_{\mathcal{CO}}$: If $i \neq j$, $\mathcal{C}$ returns x_i; otherwise, terminate with $\perp$.

Query on $\mathcal{O}_{\mathcal{S}}$: $\mathcal{A}$ sends the tuple (m, Φ, Ψ) to $\mathcal{C}$, $\mathcal{C}$ randomly chooses the index $j \in \{1, 2, \ldots, n\}$ and performs the following.

1) By using $\mathcal{O}_{\mathcal{CO}}$, if $i \neq j$, $\mathcal{C}$ runs normally as it knows the private key; if $i = j$, $\mathcal{C}$ randomly selects $t \leftarrow Z_q^*$, computes $T = tQ$.
2) By using $\mathcal{O}_{\mathcal{R}}$, for $c = \{0|1\}$, $\mathcal{C}$ computes $u_c = H(\Pi, m, T, c) = \xi_c$. Then, $\mathcal{C}$ computes $\Theta = (\xi_0 + \xi_1)Q$.
3) By using $\mathcal{O}_{\mathcal{R}}$, $\mathcal{C}$ computes $h_i = H(\Pi, m, U_i, \Phi, \Psi)$, where U_i is picked arbitrarily from $\mathbb{G}_1$, $i = 1, 2, \ldots, n$, and $i \neq j$.
4) When $i = j$ $\forall r_j \leftarrow Z_q^*$, $\mathcal{C}$ computes $U_j = r_j X_j - \sum_{i \neq j}^{n}(U_i + h_i X_i)$, $h_j = H(\Pi, m, U_j, \Phi, \Psi)$. If the tuple $(\Pi, m, U_j, \Phi, \Psi, \hat{h}_j)$ has already existed in Q_H and $h_j \neq \hat{h}_j$, return to step 1) to continue the query on $\mathcal{O}_{\mathcal{S}}$; otherwise, goto the next step.
5) $\mathcal{C}$ computes $V = (r_j + th_j)\Theta$ and then returns the signature $\sigma = (T, U_1, U_2, \ldots, U_n, V)$ to $\mathcal{A}$ as response.

TABLE II
COMPUTATIONAL COST OF BPRPDS

Entities	**Phase**	**Computational Cost**
DO	Anonymous Publishing	$\mathbb{C}_{sym}$
Blockchain	Anonymous Buying	$\mathbb{C}_{pub}$
DU	Anonymous Licence Enforcement	$n\mathbb{C}_{add} + (n+5)\mathbb{C}_{mul}$
MC	Anonymous Token Acquisition	$n\mathbb{C}_{add} + (n+3)\mathbb{C}_{mul} + 2\mathbb{C}_{par} + \mathbb{C}_{sign}$
DU CS	Anonymous Data Acquisition	$\mathbb{C}_{pub}$ $\mathbb{C}_{sym}$
DU MC	Prevention of Frame Up	$\mathbb{C}_{mul}$ $\mathbb{C}_{mul} + 2\mathbb{C}_{par}$

If $\mathcal{A}$ can effectively forge a valid signature $\sigma^* = (T^*, U_1, U_2, \ldots, U_n, V^*)$ on the tuple $(\Pi^*, m^*, \Phi^*, \Psi^*)$ which is not queried to $\mathcal{O}_\mathcal{S}$, $\mathcal{A}$ will forge another valid signature $\hat{\sigma}^* = (\hat{T}^*, U_1, U_2, \ldots, U_n, \hat{V}^*)$ with non-negligible probability according to the forking lemma of ring signature [44]. For $j \in \{1, 2, \ldots, n\}$, we have $h_j^* \neq \hat{h_j}^*$. But for any other index $i \in \{1, 2, \ldots, n\}$ and $i \neq j$, we have $h_i^* = \hat{h_i}^*$. Thus, both signatures generated by $\mathcal{A}$ satisfy the verification equation. Then, we get

$$e\Big(P, V^*\Big) = e\left(\sum_{i=1}^{n}\big(U_i + h_i^* X_i\big), \Theta^*\right) \quad (1)$$

$$e\Big(P, \hat{V}^*\Big) = e\left(\sum_{i=1}^{n}\Big(U_i + \hat{h_i}^* X_i\Big), \hat{\Theta}^*\right). \quad (2)$$

Subtract (2) from (1), we get

$$\begin{aligned} e\Big(P, V^* - \hat{V}^*\Big) &= e\Big(P, a\Big(h_j^* - \hat{h_j}^*\Big)\Big(\Theta^* - \hat{\Theta}^*\Big)\Big) \\ &= e\Big(P, a\Big(h_j^* - \hat{h_j}^*\Big)\Big(\xi_0^* - \hat{\xi_0}^* + \xi_1^* - \hat{\xi_1}^*\Big)Q\Big). \end{aligned}$$

Then, we get

$$a\Big(\xi_0^* - \hat{\xi_0}^* + \xi_1^* - \hat{\xi_1}^*\Big)Q = \Big(V^* - \hat{V}^*\Big)\Big(h_j^* - \hat{h_j}^*\Big)^{-1}.$$

$\mathcal{A}$ solves the CDHP problem with a non-negligible probability. Therefore, our scheme satisfies unforgeability under the assumption of CDHP. ■

Theorem 4 (Nonframeability): Our scheme satisfies nonframeability for any PPT adversary $\mathcal{A}$ if DLP assumption holds.

Proof: Assume $\mathcal{A}$ is playing the game with the challenger $\mathcal{C}$ in Definition 6. $\mathcal{A}$ is given the public key set $\Pi_{pk} = \{X_1, X_2, \ldots, X_n, \ldots, X_{n'}\}$ in the whole system and the message m. There exists an uncorrupted index $k \in \{1, 2, \ldots, n'\}$ such that $X_k = x_k P$. If $\mathcal{A}$ breaks the nonframeability, $\mathcal{A}$ can output a valid signature $\sigma_k = (T', U_1', U_2', \ldots, U_n', V')$ such that X_k cannot deny the signature. It demonstrates that T' and X_k are consistent in the private key x_k. $\forall d \leftarrow Z_q^*$, we can get $e(T', dP) = e(dQ, X_k)$. Then, we get $e(T', dP) = e(x_k Q, dP)$. Thus, we have $T' = x_k Q$. $\mathcal{A}$ solves the DLP problem with a non-negligible probability. Therefore, our scheme satisfies nonframeability under the assumption of DLP. ■

Theorem 5 (Behavior Profile Building Prevention): Any PPT adversary $\mathcal{A}$ can build the behavior profile of DUs with negligible probability in our BPRPDS scheme.

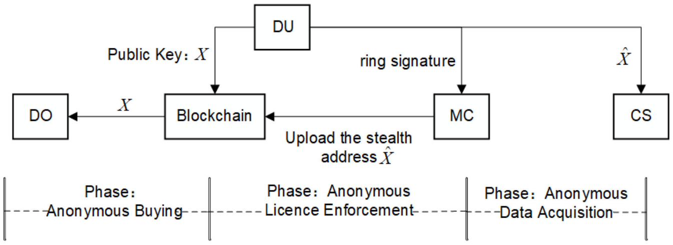

DU
Public Key：X
ring signature
$\hat{X}$
DO
X
Blockchain
MC
CS
Upload the stealth address $\hat{X}$
Phase：Anonymous Buying
Phase：Anonymous Licence Enforcement
Phase：Anonymous Data Acquisition

Fig. 4. Behavior profile building prevention of DU in different phases.

Proof: There are two types of adversaries in BPRPDS. One adversary is the external observer $\mathcal{A}_1$, and the other is the internal entity of the system $\mathcal{A}_2$. If $\mathcal{A}_1$ can learn the behavior profile of DUs, it breaks the anonymity in Theorem 2. Next, we discuss the possibility of $\mathcal{A}_2$ successfully linking public key and behavior of DU. In the phase anonymous licence enforcement, the MC can guess the identity of DU no better than $(1/n)$ based on n ring members. In the phase anonymous data acquisition, the access address observed by CS is the stealth address of DU. The stealth address is not associated with the DU's public key. It can be seen that the possibility of $\mathcal{A}_2$ building the behavior profile database of DUs is negligible. Therefore, the theorem is proved. ■

In order to clarify Theorem 5, we show the behavior profile building prevention of DU in different phases in Fig. 4.

B. Performance Evaluation

In this section, we analyze the performance of the proposed BPRPDS scheme. After that, we present the experimental results.

Assume there are n members in the ring signature group. Then, we give some denotations: the computational cost of point additive operation on $\mathbb{G}_1$ is denoted as $\mathbb{C}_{\text{add}}$, the computational cost of point multiplication on $\mathbb{G}_1$ is denoted as $\mathbb{C}_{\text{mul}}$, and the computational cost of bilinear pairing operation is denoted as $\mathbb{C}_{\text{par}}$. In addition, we denote the computational cost of the public key encryption/decryption algorithm as $\mathbb{C}_{\text{pub}}$, symmetric encryption/decryption algorithm as $\mathbb{C}_{\text{sym}}$, and signature/verification algorithm as $\mathbb{C}_{\text{sign}}$, respectively. We ignore other computational cost because they are very small.

The computational cost of each phase is shown in Table II. In the phase anonymous publishing, the computational cost of DO is $\mathbb{C}_{\text{sym}}$. In the phase anonymous buying, the

TABLE III
COMPARISON OF DIFFERENT DENIABLE RING SIGNATURE SCHEMES

No.	Schemes	Security Properties	Phases for Comparison	Total Computational Cost of Phases
1	Komano et al. [37]	- Conditional Anonymity - Traceability - Non-frameability	- Signing - Verification - Confirmation - Disavowal	$(8n+8k-1)\mathbb{C}_e$
2	Zeng et al. [43]	- Conditional Anonymity - Traceability - Non-frameability	- Signing - Verification - Confirmation - Disavowal	$(4n+1)\mathbb{C}_{add}+2n\mathbb{C}_{mul}+20\mathbb{C}_e+9\mathbb{C}_{par}$
3	Ours	- Anonymity - Profile Building Prevention - Non-frameability	- Signing - Verification - Disavowal	$2n\mathbb{C}_{add}+(2n+9)\mathbb{C}_{mul}+4\mathbb{C}_{par}$

TABLE IV
TIME COST(IN MS) OF ENCRYPTION AND DECRYPTION FOR M

Phase	Minimum Time	Maximum Time	Average Time
Encryption	70	78	74
Decryption	35	43	39

computational cost of blockchain is $\mathbb{C}_{\text{pub}}$. In the phase anonymous licence enforcement, the computational cost of DU is $n\mathbb{C}_{\text{add}} + (n+5)\mathbb{C}_{\text{mul}}$. In the phase anonymous token acquisition, the MC performs the ring signature verification and generates the stealth address for DU. The computational cost of MC is $n\mathbb{C}_{\text{add}} + (n+3)\mathbb{C}_{\text{mul}} + 2\mathbb{C}_{\text{par}} + \mathbb{C}_{\text{sign}}$. In the phase anonymous data acquisition, the computational cost of DU and CS is $\mathbb{C}_{\text{pub}}$ and $\mathbb{C}_{\text{sym}}$, respectively. In the phase prevention of frame up, the computational cost of DU and MC are $\mathbb{C}_{\text{mul}}$ and $\mathbb{C}_{\text{mul}} + 2\mathbb{C}_{\text{par}}$, respectively.

In order to demonstrate the efficiency of our scheme, our scheme BPRPDS is compared with some undeniable ring signature schemes [37], [43]. Denote the index of the framed signer in the ring group as k. Denote the computational cost of exponentiation as $\mathbb{C}_e$. In Table III, we compare the security properties of different schemes. Besides, Table III shows the total computational cost of different schemes in phases of signing, verification, confirmation, and disavowal. The computational cost of BPRPDS is obviously better than that of schemes [37], [43]. In [37] and [43], the disavowal of the framed signer must be based on the confirmation protocol of the signer. At the same time, the anonymity of the framed signer must be opened when he/she denies the signature. Therefore, our scheme is effective and attractive.

For evaluating the performance of our BPRPDS scheme, we have implemented it in the simulation environment. We utilize the java pairing-based cryptography library (JPBC) [45] to simulate the cryptographic operations. In the experiment, we pick a supersingular elliptic curve with 512-bits elements in its base field along with the size of $\mathbb{G}_1$'s group order is 160 bits. We assume the length of private data M is 256 bits. Table IV illustrates the time cost (in ms) of encryption and decryption for M. Fig. 5 depicts the time cost in phases anonymous licence enforcement and anonymous token acquisition, where the horizontal axis represents the number of members in the ring signature group and the vertical axis represents the time cost (in ms). From Fig. 5, it can be seen that the computational cost increases linearly with the growth of ring members. Fig. 6 depicts the total time cost comparison between different schemes [37], [43], where the horizontal axis represents the number of members in the ring signature group and the vertical axis represents the time cost (in ms) of different schemes. For simplicity, we set $k = 1$ in [37]. As can be seen from Fig. 6, BPRPDS has better performance.

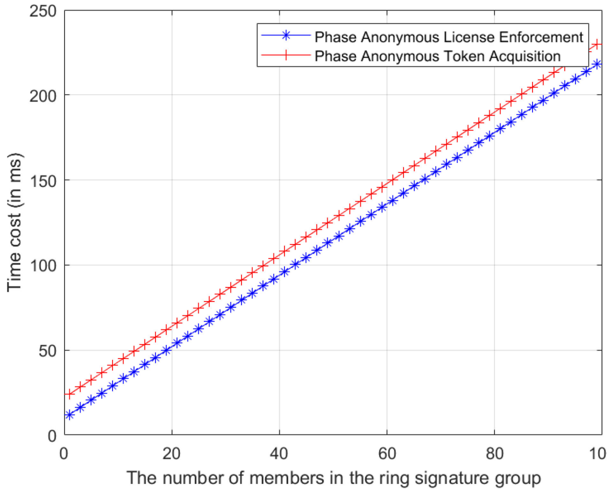


Fig. 5. Time cost of phases anonymous licence enforcement and anonymous token acquisition.

C. Applications in IoT

In the 5G era, everything is interconnected. It means that more IoT devices will be connected to the network, such as self-driving cars, UAVs, and wearable devices. Through the data sharing of these intelligent devices, the value of IoT data can be reflected. In the real IoT environment, IoT devices are limited in energy resources, computing resources, and storage resources. In addition, it is difficult for individuals to set security policies in IoT devices. These difficulties of private data sharing in the IoT environment can be solved by new technologies. In 5G, mobile edge computing (MEC) can expand the functions of IoT devices by transferring computing-intensive tasks to edge devices [46]. The end-edge-cloud cooperation

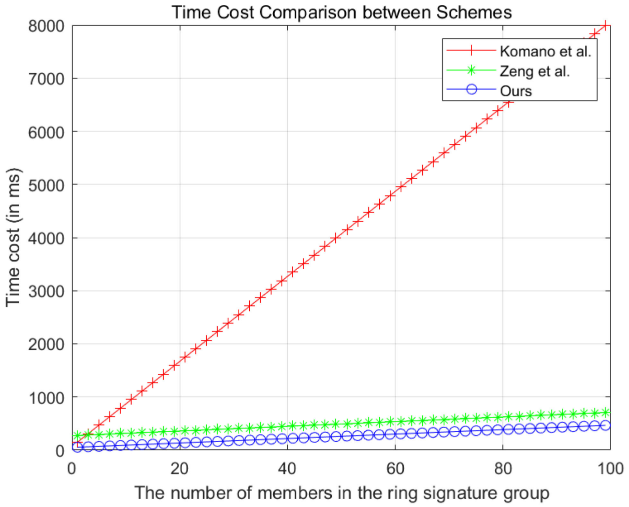


Fig. 6. Time cost comparison between schemes.

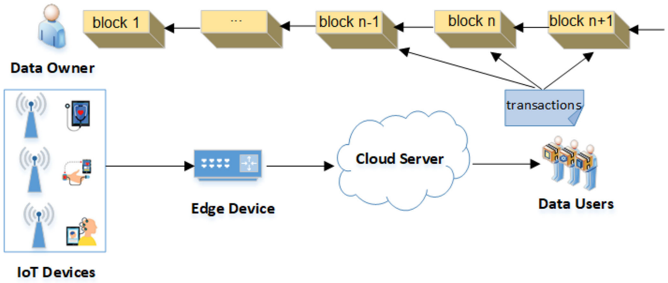
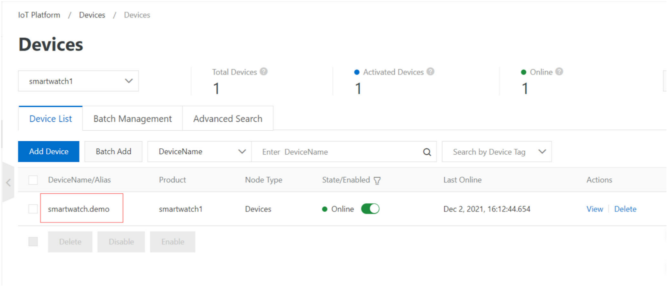


Fig. 7. Use case for private data sharing in IoT.

schemes based on the blockchain have been proposed by many researchers [47], [48].

Based on the healthcare scenario mentioned in Section I-A, we present a use case for private data sharing in IoT. In Fig. 7, the edge device helps the DO collect healthcare data from his/her IoT devices. The edge device with certain computing power can undertake the computing tasks of IoT devices (e.g., encryption). The cloud server stores shared the ciphertext of private data. The DO publishes sales rules and access policies of private data sharing by smart contracts. In the process of private IoT data sharing, smart contracts are also responsible for generating licenses for DUs. Thus, the blockchain technology enhances the interoperability of the IoT system.

Taking the smartwatch as an example, we give the specific implementation of the IoT scenario. The DO deploys the IoT platform at the edge device and uses message queuing telemetry transport (MQTT) protocol to publish and subscribe IoT data. As shown in Fig. 8, we create a smartwatch product (named smartwatch.demo) on the Alibaba IoT platform. We use the MQTT.fx (v1.7.1) tool to configure and test the uplink communication and downlink communication between the smartwatch and the cloud server. Then, we apply the Node.js (v16.13.1) tool to start the data publishing of smartwatch.demo, mainly, including three parameters: 1) height; 2) weight; and 3) blood pressure. As shown in Fig. 9, we query the plaintext data log reported by smartwatch.demo in the log service of the cloud server. In Fig. 10,


IoT Platform / Devices / Devices

Devices

smartwatch1 | Total Devices 1 | ● Activated Devices 1 | ● Online 1

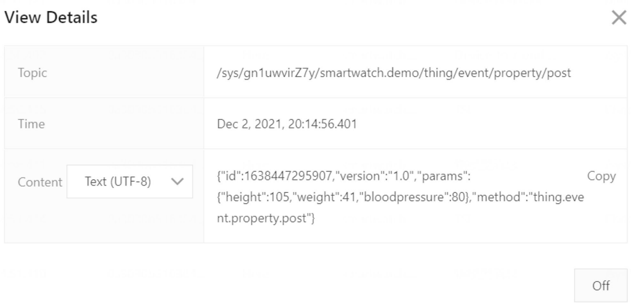
Device List | Batch Management | Advanced Search

Add Device | Batch Add | DeviceName | Enter DeviceName | Search by Device Tag

□	DeviceName/Alias	Product	Node Type	State/Enabled	Last Online	Actions
□	smartwatch.demo	smartwatch1	Devices	● Online	Dec 2, 2021, 16:12:44.654	View | Delete

□ Delete | Disable | Enable

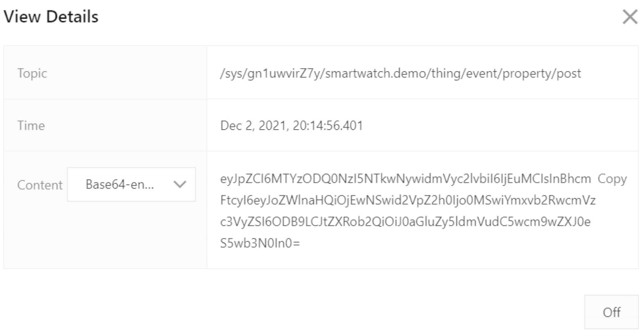
Fig. 8. Smartwatch demo for private data sharing in IoT.


View Details ×

Topic	/sys/gn1uwvirZ7y/smartwatch.demo/thing/event/property/post
Time	Dec 2, 2021, 20:14:56.401
Content Text (UTF-8)	{"id":1638447295907,"version":"1.0","params":{"height":105,"weight":41,"bloodpressure":80},"method":"thing.event.property.post"} Copy

Off

Fig. 9. Plaintext data log of the smartwatch.


View Details ×

Topic	/sys/gn1uwvirZ7y/smartwatch.demo/thing/event/property/post
Time	Dec 2, 2021, 20:14:56.401
Content Base64-en...	eyJpZCI6MTYzODQ0NzI5NTkwNywidmVyc2lvbiI6IjEuMCIsInBhcmFtcyI6eyJoZWlnaHQiOjEwNSwid2VpZ2h0Ijo0MSwiYmxvb2RwcmVzc3VyZSI6ODB9LCJtZXRob2QiOiJ0aGluZy5ldmVudC5wcm9wZXJ0eS5wb3N0In0= Copy

Off

Fig. 10. Ciphertext data log of the smartwatch.

we give the corresponding log of ciphertext data published by smartwatch.demo.

Then, we discuss how to access the blockchain network. Hyperledger Fabric 2.2 is picked as our experimental environment. On the servers over Ubuntu 18.04 OS, we install Docker-ce-19.03.4, Docker composes 1.27.4, Fabric samples 3.3.2, Go 1.15.6, and other tools. We first create a CA node, an administrator node, two client nodes, and two peer nodes on Hyperledger Fabric 2.2. As an administrator node, the MC is deployed in the membership service providers (MSPs) component. Two client nodes (i.e., the DO and DU) access to Hyperledger Fabric through the command-line interface (CLI). These nodes are joined to the created channel. Then, smart contracts are installed on client nodes. Hyperledger Fabric platforms above version 2.0 set privacy collections for users. Sensitive data in the smart contract (e.g., the encryption key K in the licence) is passed using the transient flag. In Hyperledger Fabric 2.2, transient data is transmitted as binary data. When using CLI, sensitive data must be base64 encoded and transmitted by transport layer security (TLS). As the input of transient data transfer, sensitive data will not be persisted in the transaction. After a period of time, the sensitive data will be cleared, and only the hash of the sensitive data will be retained. The sensitive data is then written to the side database on the node. In the initialization phase, a series of parameters are input through the invoke function. The client node executes the query function to obtain the licence. From the shell script, it is observed that the average running time of smart contracts is 0.68 s. Through the above analysis and performance evaluation, the proposed BPRPDS is feasible in the real IoT environment.

VI. Conclusion

In this article, we proposed an incentive private data-sharing scheme based on blockchain. Most privacy-preserving data-sharing schemes satisfy the two security attributes of anonymity and unforgeability. The proposed BPRPDS also satisfies the security attributes of nonframeability and behavior profile building prevention. Blockchain guarantees the anonymity of both sides of transactions of private data sharing. The licence generated by smart contract enables the DO to set access control policy of multisharing. With the help of deniable ring signature and Monero, DUs obtain shared data in the privacy way and can not be framed. The formal security proof shows that BPRPDS is provably secure. Performance analysis demonstrates that the BPRPDS is effective and practical. Distributed data-sharing system has been paid more and more attention on big data application. The key distribution and anonymous authentication in the data-sharing system for IoT is our future research direction.

References

[1] S. M. Errapotu, J. Wang, Y. Gong, J.-H. Cho, and M. Pan, “Safe: Secure appliance scheduling for flexible and efficient energy consumption for smart home IoT,” *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4380–4391, Dec. 2018.

[2] A. H. Sodhro, S. Pirbhulal, and A. K. Sangaiah, “Convergence of IoT and product lifecycle management in medical health care,” *Future Gener. Comput. Syst.*, vol. 86, pp. 380–391, Sep. 2018.

[3] F. Zhu, Y. Lv, Y. Chen, X. Wang, G. Xiong, and F.-Y. Wang, “Parallel transportation systems: Toward IoT-enabled smart urban traffic control and management,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4063–4071, Oct. 2020, doi: 10.1109/TITS.2019.2934991.

[4] Y. Liu, T. Dillon, W. Yu, W. Rahayu, and F. Mostafa, “Noise removal in the presence of significant anomalies for industrial IoT sensor data in manufacturing,” *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7084–7096, Aug. 2020, doi: 10.1109/JIOT.2020.2981476.

[5] J. Ni, X. Lin, and X. S. Shen, “Efficient and secure service-oriented authentication supporting network slicing for 5G-enabled IoT,” *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 644–657, Mar. 2018, doi: 10.1109/JSAC.2018.2815418.

[6] U. Satija, B. Ramkumar, and M. S. Manikandan, “Real-time signal quality-aware ECG telemetry system for IoT-based health care monitoring,” *IEEE Internet Things J.*, vol. 4, no. 3, pp. 815–823, Jun. 2017, doi: 10.1109/JIOT.2017.2670022.

[7] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, “Blockchain for the IoT and industrial IoT: A review,” *Internet Things*, vol. 10, Jun. 2020, Art. no. 100081.

[8] S. Nakamoto. “Bitcoin: A Peer-to-Peer Electronic Cash System.” 2009. [Online]. Available: http://bitcoin.org/bitcoin.pdf

[9] V. Buterin, “A next generation smart contract and decentralized application platform,” Ethereum, Zug, Switzerland, Rep., 2016. [Online]. Available: https://ethereum.org/en/whitepaper/

[10] S. Li, K. Xue, D. S. L. Wei, H. Yue, N. Yu, and P. Hong, “SecGrid: A secure and efficient SGX-enabled smart grid system with rich functionalities,” *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1318–1330, 2020, doi: 10.1109/TIFS.2019.2938875.

[11] V. Costan and S. Devadas, “Intel SGX explained,” *IACR Cryptol. ePrint Archive*, vol. 2016, no. 86, pp. 1–118, 2016. [Online]. Available: https://eprint.iacr.org/2016/086.pdf

[12] Y. Tian, Z. Wang, J. Xiong, and J. Ma, “A blockchain-based secure key management scheme with trustworthiness in DWSNs,” *IEEE Trans. Ind. Informat.*, vol. 16, no. 9, pp. 6193–6202, Sep. 2020, doi: 10.1109/TII.2020.2965975.

[13] J. Xiong, R. Bi, M. Zhao, J. Guo, and Q. Yang, “Edge-assisted privacy-preserving raw data sharing framework for connected autonomous vehicles,” *IEEE Wireless Commun.*, vol. 27, no. 3, pp. 24–30, Jun. 2020, doi: 10.1109/MWC.001.1900463.

[14] J. Xiong *et al.*, “A personalized privacy protection framework for mobile crowdsensing in IIoT,” *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4231–4241, Jun. 2020, doi: 10.1109/TII.2019.2948068.

[15] N. Van Saberhagen, “Cryptonote v 2.0,” Rep. Accessed: 2013. [Online]. Available: https://www.getmonero.org/ru/resources/research-lab/pubs/whitepaper_annotated.pdf

[16] S.-F. Sun, M. H. Au, J. K. Liu, and T. H. Yuen, “RingCT 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency Monero,” in *Proc. Eur. Symp. Res. Comput. Security*, 2017, pp. 456–474.

[17] G. Fuchsbauer, M. Orrù, and Y. Seurin, “Aggregate cash systems: A cryptographic investigation of Mimblewimble,” in *Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn.*, 2019, pp. 657–689.

[18] N. Z. Aitzhan and D. Svetinovic, “Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams,” *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 840–852, Sep./Oct. 2018.

[19] L. Li *et al.*, “CreditCoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2204–2220, Jul. 2018.

[20] H. Wang, Q. Wang, D. He, Q. Li, and Z. Liu, “BBARS: Blockchain-based anonymous rewarding scheme for V2G networks,” *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3676–3687, Apr. 2019.

[21] C. Qiu, H. Yao, F. R. Yu, C. Jiang, and S. Guo, “A service-oriented permissioned blockchain for the Internet of Things,” *IEEE Trans. Services Comput.*, vol. 13, no. 2, pp. 203–215, Mar./Apr. 2020.

[22] C. Gorenflo, S. Lee, L. Golab, and S. Keshav, “FastFabric: Scaling hyperledger fabric to 20 000 transactions per second,” *Int. J. Netw. Manag.*, vol. 30, no. 5, p. e2099, 2020.

[23] D. Huang, X. Ma, and S. Zhang, “Performance analysis of the raft consensus algorithm for private blockchains,” *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 1, pp. 172–181, Jan. 2020, doi: 10.1109/TSMC.2019.2895471.

[24] K. Liang, W. Susilo, and J. K. Liu, “Privacy-preserving ciphertext multi-sharing control for big data storage,” *IEEE Trans. Inf. Forensics Security*, vol. 10, pp. 1578–1589, 2015.

[25] X. Huang *et al.*, “Cost-effective authentic and anonymous data sharing with forward security,” *IEEE Trans. Comput.*, vol. 64, no. 4, pp. 971–983, Apr. 2015, doi: 10.1109/TC.2014.2315619.

[26] H. Wang, D. He, A. Fu, Q. Li, and Q. Wang, “Provable data possession with outsourced data transfer,” *IEEE Trans. Services Comput.*, vol. 14, no. 6, pp. 1929–1939, Nov./Dec. 2021, doi: 10.1109/TSC.2019.2892095.

[27] K. Fan, T. Liu, K. Zhang, H. Li, and Y. Yang, “A secure and efficient outsourced computation on data sharing scheme for privacy computing,” *J. Parallel Distrib. Comput.*, vol. 135, pp. 169–176, Jan. 2020.

[28] K. Xue *et al.*, “RAAC: Robust and auditable access control with multiple attribute authorities for public cloud storage,” *IEEE Trans. Inf. Forensics Security*, vol. 12, pp. 953–967, 2017.

[29] J. Xu, K. Xue, H. Tian, J. Hong, D. S. L. Wei, and P. Hong, “An identity management and authentication scheme based on redactable blockchain for mobile networks,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 6688–6698, Jun. 2020.

[30] X. Luo, K. Xue, J. Xu, Q. Sun, and Y. Zhang, “Blockchain based secure data aggregation and distributed power dispatching for microgrids,” *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5268–5279, Nov. 2021.

[31] J. Sun, H. Xiong, S. Zhang, X. Liu, J. Yuan, and R. H. Deng, "A secure flexible and tampering-resistant data sharing system for vehicular social networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12938–12950, Nov. 2020.

[32] H. Huang, P. Zhu, F. Xiao, X. Sun, and Q. Huang, "A blockchain-based scheme for privacy-preserving and secure sharing of medical data," *Comput. Security*, vol. 99, Dec. 2020, Art. no. 102010.

[33] 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, Jun. 2020, doi: 10.1109/TII.2019.2942190.

[34] J. Xu *et al.*, "Healthchain: A blockchain-based privacy preserving scheme for large-scale health data," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8770–8781, Oct. 2019.

[35] Y. Wang *et al.*, "SPDS: A secure and auditable private data sharing scheme for smart grid based on blockchain," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7688–7699, Nov. 2021, doi: 10.1109/TII.2020.3040171.

[36] B. Dan and M. Franklin, "Identity-based encryption from the weil pairing," in *Proc. Annu. Int. Cryptol. Conf.*, vol. 32, 2003, pp. 213–229.

[37] Y. Komano, K. Ohta, A. Shimbo, and S. Kawamura, "Toward the fair anonymous signatures: Deniable ring signatures," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E90-A, pp. 54–64, Jan. 2007.

[38] L. L. Win, T. Thomas, and S. Emmanuel, "Privacy enabled digital rights management without trusted third party assumption," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 546–554, Jun. 2012.

[39] Z. Zhang and L. Zhao, "A design of digital rights management mechanism based on blockchain technology," in *Proc. Int. Conf. Blockchain*, 2018, pp. 32–46.

[40] S. Rana, M. S. Obaidat, D. Mishra, S. Mukhopadhyay, and B. Sadoun, "Computational efficient authenticated digital content distribution frameworks for DRM systems: Review and outlook," *IEEE Syst. J.*, vol. 15, no. 2, pp. 1586–1593, Jun. 2021.

[41] D. Mishra, M. S. Obaidat, S. Rana, D. Dharminder, A. Mishra, and B. Sadoun, "Chaos-based content distribution framework for digital rights management system," *IEEE Syst. J.*, vol. 15, no. 1, pp. 570–576, Mar. 2021.

[42] S. Rana and D. Mishra, "Secure and ubiquitous authenticated content distribution framework for IoT enabled DRM system," *Multimedia Tools Appl.*, vol. 79, no. 7, pp. 20319–20341, 2020.

[43] S. Zeng, S. Jiang, and Z. Qin, "An efficient conditionally anonymous ring signature in the random oracle model," *Theor. Comput. Sci.*, vol. 461, no. 1, pp. 106–114, 2012.

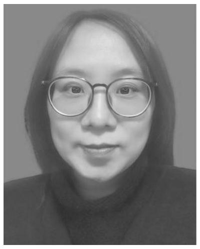
[44] J. Herranz and G. Sáez, "Forking lemmas for ring signature schemes," in *Proc. Int. Conf. Cryptol. India*, 2003, pp. 266–279.

[45] "The Java Pairing-based Cryptography Library (JPBC)." 2021. [Online]. Available: http://gas.dia.unisa.it/projects/jpbc/

[46] J. He, J. Wei, K. Chen, Z. Tang, Y. Zhou, and Y. Zhang, "Multitier fog computing with large-scale IoT data analytics for smart cities," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 677–686, Apr. 2018, doi: 10.1109/JIOT.2017.2724845.

[47] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, 3rd Quart., 2017.

[48] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "Threats to networking cloud and edge datacenters in the Internet of Things," *IEEE Cloud Comput.*, vol. 3, no. 3, pp. 64–71, May/Jun. 2016.

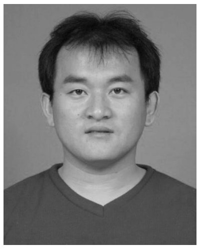


Tian Li received the M.S. degree in computer application from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2009, where she is currently pursuing the Ph.D. degree with the School of Computer Science.

Her research interests include applied cryptography and security in blockchain.

Huaqun Wang received the B.S. degree in mathematics education from Shandong Normal University, Jinan, China, in 1997, the M.S. degree in applied mathematics from East China Normal University, Shanghai, China, in 2000, and the Ph.D. degree in information security from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2006.

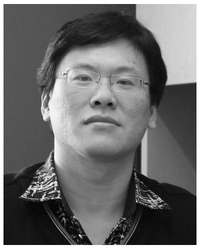
He is currently a Professor with Nanjing University of Posts and Telecommunications. He is also an expert of Shaoxing University, Shaoxing, China. His research interests include applied cryptography, network security, and cloud computing security.



Debiao He (Member, IEEE) received the Ph.D. degree in applied mathematics from the School of Mathematics and Statistics, Wuhan University, Wuhan, China, in 2009.

He is currently a Professor with the School of Cyber Science and Engineering, Wuhan University. His work has been cited more than 3400 times at Google Scholar (H-index: 32). His main research interests include cryptography and information security, in particular, cryptographic protocols.

Prof. He is an Associate Editor of *IET Wireless Sensor Systems*, *Computers & Electrical Engineering*, *Journal of Medical Systems*, and *Journal of Information Security and Applications*.



Jia Yu (Member, IEEE) received the B.S. and M.S. degrees from the School of Computer Science and Technology, Shandong University, Jinan, China, in 2000 and 2003, respectively, and the Ph.D. degree from the Institute of Network Security, Shandong University, in 2006.

He is a Professor with the College of Computer Science and Technology, Qingdao University, Qingdao, China. He was a Visiting Professor with the Department of Computer Science and Engineering, State University of New York at Buffalo, Buffalo, NY, USA, from November 2013 to November 2014. His research interests include cloud computing security, key evolving cryptography, digital signature, and network security.