Hindawi
Security and Communication Networks
Volume 2023, Article ID 9998433, 9 pages
https://doi.org/10.1155/2023/9998433

Research Article

A Generalized Blockchain-Based Government Data Sharing Protocol

Zilin Liu,[1] Anjia Yang ,[1] Huang Zeng,[1] Changkun Jiang,[2] and Li Ma[3]

[1] *College of Cyber Security, Jinan University, Guangzhou, China*
[2] *College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China*
[3] *GRGBanking Equipment Co., Ltd., Guangzhou, China*

Correspondence should be addressed to Anjia Yang; ajyang@jnu.edu.cn

Received 28 November 2022; Revised 13 January 2023; Accepted 17 January 2023; Published 6 February 2023

Academic Editor: Chien-Ming Chen

In order to catch the express train of the digital age and seize the opportunities brought by the development of blockchain technology, many government departments have begun to build blockchain-based data sharing protocols. Most existing data sharing protocols are built on different blockchains with different specific features. The interaction between them is not trivial, leading to the phenomenon of "data islands." Therefore, we consider building a data sharing protocol compatible with various blockchains. In this work, we propose a generalized blockchain-based data sharing protocol, which takes fairness, privacy, auditability, and generality into account simultaneously. With adaptor signature and zero-knowledge techniques, the proposed protocol ensures a secure and fair data sharing process and is compatible with various blockchains since it only requires the underlying blockchain to perform signature verification. Finally, we implement our construction on an Ethereum test network and conduct a series of experiments. The results demonstrate the practicality of our construction while remaining good functionalities.

1. Introduction

With the advent of digital age, the level of digital development has become one of the important indicators to measure the degree of modernization and comprehensive strength of a country. Under this trend, the construction of digital government is constantly advancing, and government data are also accumulating. In order to improve the efficiency of government departments and deepen the cooperation between departments, these data need to be shared and exchanged between government departments [1, 2]. However, due to the management and division of labor of government departments for a long time, the data of different government departments are often controlled by themselves. These data are widely stored in different units, departments, and network environments. It is difficult for them to be shared between departments, resulting in the phenomenon of "data islands."

At the same time, with properties of decentralization, traceability, and immutability, blockchain technology has been widely explored to promote data trust, sharing, co-governance, and co-maintenance [3–12]. With the help of blockchain technology, the trusted third party in the data sharing protocol is replaced by a public ledger maintained by all users. Any data sharing behavior between users will be recorded in this ledger to facilitate subsequent auditing and tracking.

However, most existing blockchain-based data sharing protocols require the underlying blockchains to provide some specific functionality. In particular, some data sharing protocols (e.g., [13, 14]) are built on blockchains, which support hash time-lock contract (HTLC). Some protocols (e.g., [15, 16]) are built relying on unspent transaction output (UTXO) structure-based blockchains. Some protocols (e.g., [4, 11, 17, 18]) are constructed on blockchains which provide smart contract functionality. Some protocols (e.g., [19–21]) are built based on the Internet of things application-based (IOTA-based) blockchain. Some protocols (e.g., [22–24]) are constructed relying on the permissioned blockchain. It is difficult for data to be shared between different blockchains with different features. In order to step over this barrier, Jiang et al. [25] proposed a new blockchain named PolyChain, which provides high modularity, flexibility, scalability, reliability, and security. Then, they create a data sharing protocol based on PolyChain. Nevertheless, their protocol requires a major update to the existing blockchain architecture, which is to migrate the protocol from the original underlying blockchain to PolyChain.

With the above consideration, we are motivated to consider whether we can construct a data sharing protocol assuming only the bare minimal ability of the underlying blockchain to verify a signature. It would be compatible with a wide variety of blockchains and does not require updates to the existing underlying architecture of blockchain-based data sharing protocols. In this work, we propose a positive answer to the above consideration. We introduce a new data sharing protocol, which takes fairness, privacy protection, auditability, and generality into account simultaneously. The contributions of our work can be summarized as follows:

(i) We devise a data sharing protocol that is compatible with various existing blockchains. The generalized data sharing protocol is constructed relying on the adaptor signature. It provides fairness, privacy protection, auditability, and generality at the same time. In this protocol, the on-chain operation is only to verify a signature, which is compatible with various non-/Turing-complete blockchains. Furthermore, existing data sharing protocols can be easily converted to generalized protocols with only a software update.

(ii) We implement our construction on an Ethereum test network [26]. We perform a series of experiments to show the effectiveness and efficiency of our construction. The results show that they require at most 35 milliseconds to be computed and a communication overhead of less than 300 bytes in the worst case. Therefore, our construction can be regarded as a promising tool to realize a practical data sharing protocol.

The remainder of the paper is organized as follows. We introduce the related work in Section 2 and present preliminaries in Section 3. Then we will discuss our system model, threat model and design goal in Section 4. Next, we illustrate our construction in Section 5 and analyze the security of our construction in Section 6. We conduct a series of experiments in Section 7. Finally, we conclude our work in Section 8.

2. Related Work

Most existing blockchain-based data sharing protocols constructed rely on some specific functionality of the underlying blockchain. Some data sharing protocols work with blockchains which support hash time-lock contract (HTLC). Mohanty et al. [13] propose SIoVChain, a time-lock contract-based privacy preserving data sharing scheme with incentives for social Internet of vehicles. Zhang et al. [14] develop a homomorphic hashing-based transaction segmentation scheme and propose an efficient IoT data sharing approach relying on it. Some protocols rely on unspent transaction output (UTXO) structure-based blockchains. Wang et al. [15] construct a novel lightweight authentication and a UTXO-based blockchain data trading system to facilitate online data trading. Noh et al. [16] built a blockchain-based user-centric records management system, which is convenient for sharing of medical records among institutions. Some protocols require the underlying blockchain to support smart contracts. Jaiman and Urovi [4] present a blockchain-based data sharing consent model which helps individuals exercise access control over health data. Shrestha et al. [27] introduce a traceable and incentive customer data sharing platform, which allow users sharing their data with business enterprises. Xu et al. [17] propose a blockchain-based secure data sharing platform with fine-grained access control. Naz et al. [11] construct a secure data sharing platform with the help of blockchain technology and an interplanetary file system. Hoang et al. [18] propose a privacy preserving blockchain-based data sharing platform that protects user anonymity and data confidentiality. Singh et al. [28] design a secure cross-domain blockchain-based data sharing protocol for IoT industrial. Theodouli et al. [29] devise a blockchain-based system to facilitate healthcare data sharing. Some protocols are based on the Internet of things application-based (IOTA-based) blockchain. Hassija et al. [19] introduce a blockchain-based framework for lightweight data sharing and energy trading. Gangwani et al. [20] facilitate data sharing among Internet of things devices with the help of IOTA-based blockchain. Abdullah et al. [21] emphasize secure sharing of health data in the digital healthcare system by exploring the potential of a IOTA Tangle [30]. Some protocols are based on the permissioned blockchain. Xia et al. [23] construct a blockchain-based data sharing for electronic medical records by utilizing the permissioned blockchain technology. Wang et al. [22] propose a medical data sharing platform based on permissioned blockchains. Xiao et al. [24] introduce a cross-organizational medical data sharing framework with the help of a permissioned blockchain.

Among them, to construct a general data sharing protocol, Jiang et al. [25] propose a new blockchain named PolyChain, which provides high modularity, flexibility, scalability, reliability, and security. Then they construct a data sharing platform based on PolyChain. However, this scheme focuses on constructing a new blockchain and designing a data sharing platform based on it but does not consider devising a data sharing platform, which is compatible for existing blockchains. Such a solution requires developers to make changes to the underlying blockchain architecture of the existing data sharing platforms, which is complicated and cumbersome to operate.

3. Preliminaries

In this section, we will cover some notations and preliminaries instructions that will be used when constructing our protocol.

3.1. Notations. The security parameter is denoted by 1^{λ} for $\lambda \in \mathbb{N}$. The notation $x \overset{\$}{\longleftarrow}$ is used to represent that an element x is uniformly sampled from a set X. We denote by $y \longleftarrow F(x)$ the output of the probabilistic polynomial time (PPT) algorithm F on input x. If the algorithm F is a deterministic polynomial time (DPT) algorithm, the notation is $y \coloneqq F(x)$.

3.2. Noninteractive Zero-Knowledge. An NP relation is denoted by R and a set of positive instances with related to the relation R is denoted by L (i.e, $L = \{x | \exists\, w s.t. R(x, w) = 1\}$). If R is poly-time decidable and for any PPT adversaries $\mathscr{A}$, the probability of $\mathscr{A}$ producing a witness w that satisfies $R(x, w) = 1$ is bounded by a negligible function; we name R as a hard relation. A noninteractive zero-knowledge proof scheme NIZK includes two algorithms. P_{NIZK} is the prover algorithm where $\pi \longleftarrow P_{\text{NIZK}}(x, w)$ and V_{NIZK} is the verifier algorithm where $\{0, 1\}\colon = V_{\text{NIZK}}(x, \pi)$. We refer readers to [31] for further details.

3.3. Adaptor Signature. An adaptor signature is defined relying on a digital signature Σ_{DS} and a hard relation R which includes four algorithms $\Sigma_{\text{AS}} = (\text{PreSig}, \text{PreVf}, \text{Adapt}, \text{Ext})$. We can presign some information over a hard relation with $\widehat{\sigma} \longleftarrow \text{PreSig}(\text{sk}, m, Y)$. The validation of the pre-signature can be verified using $\text{PreVf}(m, Y, \widehat{\sigma})$. Besides, the pre-signature can be converted to a valid signature with the corresponding witness with $\sigma\colon = \text{Adapt}(\widehat{\sigma}, y)$. With the pre-signature/signature pair $(\widehat{\sigma}, \sigma)$, we can extract the witness y using $y \coloneqq \text{Ext}(\sigma, \widehat{\sigma}, Y)$. A secure adaptor signature provides three properties: pre-signature correctness, pre-signature adaptability, and witness extractability. For more details, we refer readers to [32].

3.4. Blockchains. As in [33–35], we model an ideal ledger (blockchain) functionality $\mathbb{B}$ as a trusted append-only bulletin board. The ideal functionality $\mathscr{F}_{\mathbb{B}}$ maintains the list of all transactions of each user and updates when a new transaction is performed. It provides these properties: (a) complete decentralization, namely, the public ledger is maintained decentralized; (b) correctness and traceability, which means that each node is able to trace and verify the correctness of the data; (c) immutability implies that the on-chain transactions are tamper-resistant; and (d) cryptography, namely, the security of the blockchain is guaranteed by the underlying cryptography techniques.

4. System Model, Threat Model, and Design Goal

In this section, we will introduce the system model, the threat model, and design goals of our construction.

4.1. System Model. As shown in Figure 1, there are three roles in our system: the sender, the receiver, and the blockchain.

4.1.1. Sender. The sender is the person who wishes to transfer his data access right to someone else. He wants to transfer his data access key to the receiver in a secure way and can record the process on the blockchain.

4.1.2. Receiver. The receiver is the person who aims to receive the data access right.

4.1.3. Blockchain. The blockchain acts as an important part in our construction. It is responsible for accepting the transaction submitted by the sender and verifying the signature of the transaction.

In our system model, assume that the sender wants to grant her data access key to the receiver. She first generates a transaction, presigns it, and sends it to the receiver. The pre-signature attached to the transaction is similar to a lock. If the sender decides to grant her key, she will sign the transaction and upload it on the blockchain. Once the receiver observes that the transaction has been uploaded on chain, he can combine the previously obtained transaction with the on-chain transaction to extract the secret key and gain the access to the data.

4.2. Threat Model. We define the following security assumptions to describe the attacks our construction will be exposed.

(i) No party is trustworthy

(ii) All parties are rational and greedy, they will choose actions that are in their interests at all costs

(iii) The underlying blockchain is secure and cannot be controlled by any malicious parties

Besides, we also present some types of threats that our construction will face.

4.2.1. Breaking Data Confidentiality. Malicious parties who do not participate in the transaction can attempt to obtain the data access right by analyzing the transactions on the blockchain.

4.2.2. Breaking Fairness. A malicious sender can attempt to generate an invalid presignature over their transaction, so as to avoid the receiver obtaining the data access key in the subsequent operation.

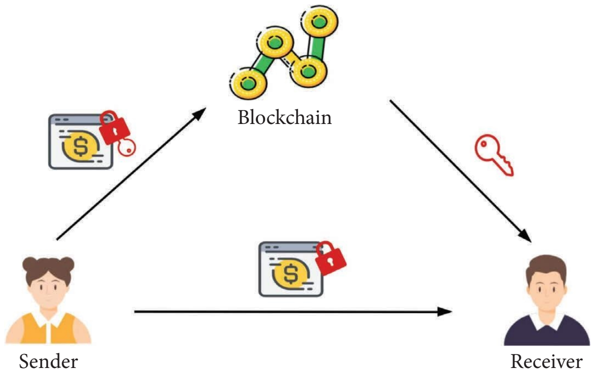

Figure 1: The system model.

4.3. Design Goal. Relying on the aforementioned system model and threat model, we define our design goals as follows:

(i) Fairness: neither party to the transaction can disrupt the execution of the construction and damage the fairness of the other party

(ii) Malicious behaviour resistance: our construction has the ability to resist the aforementioned threats

5. Our Construction

In this section, we first discuss the design challenges we need to overcome when devising our construction, then we will illustrate the proposed construction in detail. Finally, we will present a concrete instance to show how to instantiate it.

5.1. Design Challenges. It is not trivial to put forward a generalized and secure data sharing protocol. Many challenges arise when we are trying to construct such a data sharing protocol.

(i) Generality: With the development of the blockchain technology, many government departments have begun to build blockchain-based applications. In order to facilitate the data sharing between different blockchain-based applications, we are demanded to devise a generalized data sharing protocol. In this work, we utilize the adaptor signature to achieve this property. With this tool, our data sharing protocol only requires the underlying blockchains to provide signature verification and can be applied to various blockchains.

(ii) Auditability: The transfer of the data access rights between different government departments needs to be recorded in a log to facilitate subsequent audits. In this work, we leverage the blockchain technology to record it. The immutability feature of the blockchain technology can help us achieve security monitoring of data access rights and tracking of the transfer process.

(iii) Privacy: Privacy protection [36, 37] is also taken into account when transferring data access rights between different government departments. They do not want others to extract data access keys by analyzing transactions on the blockchain. In this work, our construction only requires signature verification on the blockchain, and no additional operations are required. Therefore, the malicious party cannot obtain more information than the transaction signature, which prevents him from further analysis of the transactions and protects the users' privacy.

5.2. The Generic Construction. Our generic data sharing protocol is denoted by Π. It can be achieved by leveraging an adaptor signature $\Sigma_{AS} = (\text{PreSig}, \text{PreVf}, \text{Adapt}, \text{Ext})$ relying on a digital signature $\Sigma_{DS} = (\text{Gen}, \text{Sign}, \text{Vrfy})$ which is used by the underlying blockchain and a hard relation R. The operations of our protocol can be divided into two aspects: off-chain operations and on-chain operations. The concrete details are depicted in Figure 2. We will discuss each aspect separately at a high level in the following:

5.2.1. Off-Chain Operations. Assume that government departments want to work together, department employees need to share some data during this period. These data were stored in various departments and can be accessed by those with certain permissions. In order to ensure the security of the data sharing process, the sender of the data hopes that no third party other than the receiver of the data can spy on the data access key through their operations. The sender and the receiver are denoted by U_0 and U_1, respectively. The sender firstly generates a public/private key pair (pk, sk) and sets it as a public parameter/trapdoor pair (pp, td). This pair is used to blind the information to be transmitted next, preventing others from analyzing the transmitted information to obtain some secrets (e.g., data access keys). Then, U_0 generates a puzzle P over the data access key k with the public parameter pp. The corresponding zero-knowledge proof over the puzzle P is attached to prove that the puzzle is solvable. At this moment, he generates a transaction tx which is used to pass the data access key to the receiver and presigns this transaction. Note that the pre-signature $\widehat{\sigma}$ is bound with a puzzle P, which means that the pre-signature can only be turned into a valid signature if the solution of the puzzle P is known. After receiving such a tuple from the sender, the receiver will verify the validation of the proof π and the pre-signature $\widehat{\sigma}$ to ensure that the data access key can be obtained after a validly signed transaction is uploaded on the blockchain.

5.2.2. On-Chain Operations. If the sender decides to grant the data access key to the receiver, he will sign the transaction tx and upload it on the blockchain $\mathbb{B}$ with the corresponding valid signature. If the receiver notices that the transaction tx has been uploaded on the blockchain $\mathbb{B}$, he

```
The Data Sharing Protocol $\Pi$
Off-chain operations
1) User $U_0$ does the following:
    • Calculate $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.
    • Set $(\mathsf{pp}, \mathsf{td}) := (\mathsf{pk}, \mathsf{sk})$.
    • Generate a puzzle using $P \leftarrow \mathsf{Enc}(\mathsf{pp}, k)$.
    • Calculate $\pi \leftarrow \mathsf{P}_{\mathsf{NIZK}}(\{\exists\alpha|\mathsf{Dec}(\mathsf{td}, P) = \alpha\}, \alpha)$.
    • Generate a transaction $tx$.
    • Compute $\hat{\sigma} \leftarrow \Sigma_{\mathsf{AS}}.\mathsf{PreSig}(\mathsf{sk}_0, tx, P)$.
    • Send $(P, \pi, tx, \hat{\sigma})$ to $U_1$.
2) User $U_1$ does the following:
    • Verify whether $\mathsf{V}_{\mathsf{NIZK}}(P, \pi) = 1$, otherwise abort.
    • Verify whether $\Sigma_{\mathsf{AS}}.\mathsf{PreVf}(tx, P, \hat{\sigma}) = 1$, otherwise abort.

On-chain Operations
1) User $U_0$ does the following:
    • Sign the transaction using $\sigma \leftarrow \Sigma_{\mathsf{DS}}.\mathsf{Sign}(\mathsf{sk}_0, tx)$.
    • Upload $(tx, \sigma)$ on the blockchain $\mathbb{B}$.
2) If transaction $tx$ is on chain, user $U_1$ does the following:
    • Compute $k \leftarrow \Sigma_{\mathsf{AS}}.\mathsf{Ext}(\sigma, \hat{\sigma}, P)$.
```

FIGURE 2: Algorithms and protocols for the generic construction.

can extract the key k from the pre-signature/signature pair $(\widehat{\sigma}, \sigma)$. Therefore, the receiver gains the data access and is able to read the data. The sender and the receiver successfully achieved data sharing between them.

5.3. Schnorr-Based Instance. In the following, we introduce a concrete instance Π_{Sch} to show how to instantiate our protocol. The instance is based on a Schnorr-based adaptor signature and includes two aspects: the off-chain operations and the on-chain operations. The details are shown in Figure 3.

5.3.1. Off-Chain Operations. We consider a scenario that the sender U_0 and the receiver U_1 work in different government departments, and the two government departments need to work together, so U_0 and U_1 want to share data between them. Besides, U_0 hopes to transfer a data access key k to U_1 without leaking information to the third party. In order to achieve this goal, they first mutually agree on a transaction and a corresponding pre-signature, which is used to secretly transmit the key in the latter. Note, that in this concrete construction, $\mathbb{G}$ is an elliptic curve group of prime order q with a generator g. The commitment scheme and the non-interactive zero-knowledge scheme are denoted by com and NIZK, respectively. In the beginning, U_0 and U_1 are required to generate a shared Schnorr public key pk_{01}^{Σ}. The shared key generation can be achieved by using [34]. Then, U_0 generates a puzzle P over the access key k and calculates the corresponding proof π. U_1 will verify whether the puzzle P is solvable after the puzzle/proof pair (P, π) is sent by U_0. After that, U_0 and U_1 jointly process a coin tossing protocol to agree on a randomness $R = k + r_1 + r_2$. They exchange of g^{r_1} and g^{r_2} with each other to blind the puzzle P. The corresponding proof is attached during the coin tossing process. At this moment, U_0 and U_1 can mutually calculate an "almost" valid signature $\widehat{\sigma}{:}\ = (e, \widehat{s})$ while the valid form is $(e, \widehat{s} + k)$. The pre-signature $\widehat{\sigma}$ is calculated over some messages (e.g., the transaction id) which are jointly agreed before. Notice that U_1 is able to verify the validation of such a pre-signature $\widehat{\sigma}$ during the former calculation. At this moment, they complete the off-chain operations together, and U_1 is guaranteed to obtain the data access key k if U_0 uploads the transaction attached with the valid signature on the blockchain in the latter.

5.3.2. On-Chain Operations. If the sender decides to grant the data access key k to the receiver, he will convert the "almost" valid signature to a valid signature. Please note that the valid form of the signature is $\sigma{:}\ = (e, \widehat{s} + k)$. Then, U_0 uploads the transaction on chain with the corresponding signature σ. After the transaction is recorded on the blockchain $\mathbb{B}$, U_1 can achieve the valid signature σ from the on-chain transaction. Therefore, he has the ability to combine the pre-signature $\widehat{\sigma}$ with the valid signature σ to calculate the data access key k. At this moment, the receiver can read the corresponding data with the extracted key. The corresponding data are successfully shared between the sender and the receiver. It is worth mentioning that the process of transferring data access rights will also be recorded on the blockchain, which facilitates subsequent auditing.

6. Security Analysis

We will discuss the security of the proposed construction in this section. The construction relying on adaptor signature and non-interactive zero-knowledge proof defeat against the threats is defined in Section 4. The details will be introduced in the following.

6.1. Breaking Data Confidentiality. A malicious party may attempt to extract the data access key through observing the transaction on the blockchain. Through analyzing the transactions, he can obtain a valid signature. We denote such a signature as σ. If the malicious party wants to extract the corresponding data access key k of the signature σ, he needs to acquire the corresponding pre-signature $\widehat{\sigma}$. However, the communication between the sender and the receiver in the off-chain operations takes place in a secure communication channel. Therefore, the malicious party cannot capture more information. The only way for him to obtain a pre-signature is to forge one. If he can extract the data access key from the signature and forge the pre-signature pair $(\sigma, \widehat{\sigma}')$, it means that $\widehat{\sigma}'$ is a valid forgery. In other words, the malicious party has the ability to break the unforgeability feature of the adaptor signature. The probability of such an event occurring is bounded by a negligible function, so the threat does not occur.

6.2. Breaking Fairness. A malicious sender may attempt to send an invalid pre-signature to the receiver in the off-chain operations, so as to avoid the receiver from obtaining the data access key in the on-chain operations. This threat occurs in the following situation. The malicious sender uploads a

The Schnorr-based Construction Π_{Sch}

Public parameters: group description $(\mathbb{G}, g, q)$, message m

Off-chain operations

```
$\mathsf{Off\text{-}chain}_{U_0}(k, \mathsf{sk}_0^{\Sigma}, \mathsf{pk}_{01}^{\Sigma}, m)$ :
    $r_2 \overset{\$}{\leftarrow} \mathbb{Z}_q$
    $R_2 \leftarrow g^{r_2}; P \leftarrow g^k$
    $\pi \leftarrow \mathsf{P}_{\mathsf{NIZK}}(k, \{\exists k \text{ s.t. } P = g^k\})$
    $\pi_2 \leftarrow \mathsf{P}_{\mathsf{NIZK}}(r_2, \{\exists r_2 \text{ s.t. } R_2 = g^{r_2}\})$
    $(\mathsf{com}, \mathsf{decom}) \leftarrow \mathsf{P}_{\mathsf{com}}((R_2, \pi_2))$

                        $\xrightarrow{P,\pi,\mathsf{com}}$

$\mathsf{Off\text{-}chain}_{U_1}(\mathsf{sk}_1^{\Sigma}, \mathsf{pk}_{01}^{\Sigma}, m)$ :
    If $\mathsf{V}_{\mathsf{NIZK}}(P, \pi) \neq 1$ then abort
    $r_1 \overset{\$}{\leftarrow} \mathbb{Z}_q; R_1 \leftarrow g^{r_1}$
    $\pi_1 \leftarrow \mathsf{P}_{\mathsf{NIZK}}(r_1, \{\exists r_1 \text{ s.t. } R_1 = g^{r_1}\})$

                        $\xleftarrow{R_1,\pi_1}$

U_0:
    If $\mathsf{V}_{\mathsf{NIZK}}(R_1, \pi_1) \neq 1$ then abort
    $R \leftarrow R_1 \cdot R_2 \cdot P$
    $e := H(R || \mathsf{pk}_{01}^{\Sigma} || m)$
    $s_2 \leftarrow r_2 - \mathsf{sk}_0^{\Sigma} \cdot e \bmod q$

                        $\xrightarrow{\mathsf{decom},R_2,\pi_2,s_2}$

U_1:
    If $\mathsf{V}_{\mathsf{com}}(\mathsf{com}, \mathsf{decom}, (R_2, \pi_2)) \neq 1$ then abort
    If $\mathsf{V}_{\mathsf{NIZK}}(R_2, \pi_2) \neq 1$ then abort
    $R \leftarrow R_1 \cdot R_2 \cdot P$
    $e := H(R || \mathsf{pk}_{01}^{\Sigma} || m)$
    If $g^{s_2} \neq R_2 \cdot (Q/g^{\mathsf{sk}_1^{\Sigma}})^{-e}$ then abort
    $s_1 \leftarrow r_1 - \mathsf{sk}_1^{\Sigma} \cdot e \bmod q$
    $\hat{s} \leftarrow s_1 + s_2 \bmod q$

                        $\xleftarrow{\hat{s}}$

U_0:
    If $g^{\hat{s}} \neq R_1 \cdot R_2 \cdot (\mathsf{pk}_{01}^{\Sigma})^{-e}$ then abort
    $\hat{\sigma} := (e, \hat{s})$
    return $\hat{\sigma}$

U_1:
    $\hat{\sigma} := (e, \hat{s})$
    return $\hat{\sigma}$
```

On-chain operations

```
$\mathsf{On\text{-}chain}_{U_0}(\mathsf{sk}_0^{\Sigma}, \mathsf{pk}_{01}^{\Sigma})$ :
    $s \leftarrow \hat{s} + k$
    If verification of $(e, s)$ fails then abort
    Else publish signature $(e, s)$ on chain
    return $\top$

$\mathsf{On\text{-}chain}_{U_1}(\mathsf{sk}_1^{\Sigma}, \mathsf{pk}_{01}^{\Sigma})$ :
    $k \leftarrow s - \hat{s}$
    return $k$
```

Figure 3: The Schnorr-based construction.

Table 1: The resources required to execute the algorithms for our construction.

		Sender	Receiver
Off-chain	Time (ms)	27.66	30.78
	Comm (bytes)	272	94
On-chain	Time (ms)	7.11	0.006
	Comm (bytes)	—	—
Total	Time (ms)	34.77	30.79
	Comm (bytes)	272	94
Computation cost (gas)		45734	0

transaction with a valid signature σ' on the blockchain, while the receiver cannot extract a valid witness from it. If this is the case, it means that we cannot extract a valid witness from the valid pre-signature and signature pair $(\widehat{\sigma}, \sigma')$. That is to say, the malicious sender is able to against the witness extractability feature of the adaptor signature. The probability of such a situation occurring is bounded by a negligible function. Therefore, the threat cannot happen.

7. Performance Analysis

In order to demonstrate the feasibility and the performance of our construction, we develop a prototypical Python implementation and conduct a series of experiments. We instantiate the Schnorr signature over the elliptic curve secp256k1 (the one used in Bitcoin), use the libraries math, and random for corresponding calculations. The commitment scheme is modeled as a random oracle [38] with the SHA-256 algorithm.

7.1. Testbed. We conduct our experiments on a personal computer with an Intel (R) Core (TM) i7-10875H, 2.30 GHz, and 32 GB RAM. We measure the algorithms in on-chain and off-chain operations but do not consider the key generation algorithm for that this phase does not affect the online performance of our construction. We refer readers to [39] for the details of the key generation algorithm. The details of our evaluation is depicted in Table 1.

7.2. Computation Time. We measure the computation time that each user required when performing different algorithms. From Table 1, we observe that the computation time required for the off-chain operations accounted for a large portion of the total time. Notice that in this protocol, we grant the data access key on the blockchain. Therefore, the computation time does not vary with the size of the data. The time spent on off-chain operations accounts for the majority of each user's operational time since the sender and the receiver need to perform verification to prove they behave honestly. In particular, it only takes 0.006 ms for the receiver to obtain his data access key in the on-chain operations.

7.3. Communication Overhead. We measure the communication overhead by calculating the information that the sender and the receiver need to exchange during the execution of each phase. The communication overhead is mainly generated in the off-chain operations. The sender need to send 272 bytes of information to the receiver and the receiver responds with 94 bytes information. The communication of the sender is higher than that of the receiver since the senders needs to send some additional proof information. In addition, there is a communication overhead for the receiver because the two parties involved need to perform a coin tossing protocol to jointly generate the relevant transaction information.

7.4. Computation Cost. We implement our construction on an Ethereum test network [26] and measure the computation cost through calculating the gas required by the smart contract. We observe that 45734 units of gas is required when the sender conducts his transaction on chain. At the time of writing, we consult the Ethereum gas price website "ETH Gas Station" [40] to know that the average gas price per unit is 14.9 gwei; our construction therefore costs considerably less than 0.0007 Ether.

8. Conclusion

In this work, we devise a blockchain-based data sharing protocol, which takes fairness, privacy protection, auditability, and generality into account simultaneously. Besides, we show how to instantiate it by presenting a Schnorr-based instance. Finally, we conduct a series of experiments to illustrate the effectiveness and efficiency of our construction. The results show that our construction is piratical and can be regarded as a promising tool to realize a generalized and secure data sharing platform.

Data Availability

No data were used to support the results of our study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was funded by the Key-Area Research and Development Program of Guangdong Province (Grant nos. 2020B0101090004 and 2020B0101360001), the National Key R&D Program of China (2020YFB1005600 and 2021ZD0112802), the National Natural Science Foundation of China (62072215, 61825203, and U1736203), the Major

Program of Guangdong Basic and Applied Research Project (2019B030302008), and the Guangdong Provincial Science and Technology Project (2019B010137002 and 2017B010111005).

References

[1] L. Zhou, R. Huang, and B. Li, ““What is mine is not thine”: understanding barriers to China’s interagency government data sharing from existing literature,” *Library & Information Science Research*, vol. 42, no. 3, Article ID 101031, 2020.

[2] Asli Yagmur Akbulut-Bailey, “Information sharing between local and state governments,” *Journal of Computer Information Systems*, vol. 51, no. 4, pp. 53–63, 2011.

[3] M. Li, J. Weng, A. Yang et al., “Crowdbc: a blockchain-based decentralized framework for crowdsourcing,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 6, pp. 1251–1266, 2019.

[4] V. Jaiman and V. Urovi, “A consent model for blockchain-based health data sharing platforms,” *IEEE Access*, vol. 8, pp. 143734–143745, 2020.

[5] T. Li, A. Yang, J. Weng, Y. Tong, and Q. Pei, “Concurrent and efficient iot data trading based on probabilistic micropayments,” *Wireless Networks*, vol. 29, no. 2, pp. 607–622, 2022.

[6] Z. Liu, A. Yang, J. Weng, T. Li, H. Zeng, and X. Liang, “Gmhl: generalized multi-hop locks for privacy-preserving payment channel networks,” *Cryptology ePrint Archive*, Report 2022/115, https://ia.cr/2022/115, 2022.

[7] F. Antonucci, S. Figorilli, C. Costa, F. Pallottino, L. Raso, and P. Menesatti, “A review on blockchain applications in the agri-food sector,” *Journal of the Science of Food and Agriculture*, vol. 99, no. 14, pp. 6129–6138, 2019.

[8] X. Liu, A. Yang, C. Huang, Y. Li, T. Li, and M. Li, “Decentralized anonymous authentication with fair billing for space-ground integrated networks,” *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 7764–7777, 2021.

[9] A. Yang, J. Weng, K. Yang, C. Huang, and X. Shen, “Delegating authentication to edge: a decentralized authentication architecture for vehicular networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1284–1298, 2022.

[10] M. Li, J. Weng, A. Yang, J.-N. Liu, and X. Lin, “Toward blockchain-based fair and anonymous ad dissemination in vehicular networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11248–11259, 2019.

[11] M. Naz, F. A. Al-zahrani, R. Khalid et al., “A secure data sharing platform using blockchain and interplanetary file system,” *Sustainability*, vol. 11, no. 24, p. 7054, 2019.

[12] B. Mi, B. Wu, D. Huang, Y. Liu, L. Chen, and S. Wan, “Privacy-oriented transaction for public blockchain via secret sharing,” *Security and Communication Networks*, vol. 2022, Article ID 9946088, 19 pages, 2022.

[13] S. K. Mohanty and S. Tripathy, “Siovchain: time-lock contract based privacy-preserving data sharing in siov,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24071–24082, 2022.

[14] Y. Zhang, K. Gai, J. Xiao, L. Zhu, K. K. R. Choo, and Kwang Raymond Choo, “Blockchain-empowered efficient data sharing in internet of things settings,” *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 12, pp. 3422–3436, 2022.

[15] F. Wang, L. Cheng, P. She, and M. Huang, “Redats: a restful data trading system with blockchain,” in *Proceedings of the 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA)*, pp. 589–593, Shenyang, China, January 2021.

[16] S.-W. Noh, Y. Park, C. Sur, S. U. Shin, and K. H. Rhee, “Blockchain-based user-centric records management system,” *International Journal of Control and Automation*, vol. 10, no. 11, pp. 133–144, 2017.

[17] H. Xu, Q. He, X. Li, B. Jiang, and K. Qin, “Bdss-fa: a blockchain-based data security sharing platform with fine-grained access control,” *IEEE Access*, vol. 8, pp. 87552–87561, 2020.

[18] V.-H. Hoang, E. Lehtihet, and Y. Ghamri-Doudane, “Privacy-preserving blockchain-based data sharing platform for decentralized storage systems,” in *Proceedings of the 2020 IFIP Networking conference (networking)*, pp. 280–288, Paris, France, June 2020.

[19] V. Hassija, V. Chamola, S. Garg, D. N. G. Krishna, G. Kaddoum, and D. N. K. Jayakody, “A blockchain-based framework for lightweight data sharing and energy trading in v2g network,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5799–5812, 2020.

[20] P. Gangwani, A. Perez-Pons, T. Bhardwaj, H. Upadhyay, S. Joshi, and L. Lagos, “Securing environmental iot data using masked authentication messaging protocol in a dag-based blockchain: iota tangle,” *Future Internet*, vol. 13, pp. 312–12, 2021, https://www.mdpi.com/1999-5903/13/12/312.

[21] S. Abdullah, J. Arshad, M. M. Khan, M. Alazab, and K. Salah, “Prised tangle: a privacy-aware framework for smart healthcare data sharing using iota tangle,” *Complex & Intelligent Systems*, vol. 1–19, 2022.

[22] R. Wang, W.-T. Tsai, J. He, C. Liu, Q. Li, and E. Deng, “A medical data sharing platform based on permissioned blockchains,” in *Proceedings of the 2018 International Conference on Blockchain Technology and Application*, pp. 12–16, Xi’an, China, December 2018.

[23] Q. Xia, E. Sifah, A. Smahi, S. Amofa, and X. Zhang, “Bbds: blockchain-based data sharing for electronic medical records in cloud environments,” *Information*, vol. 8, pp. 44–52, 2017, https://www.mdpi.com/2078-2489/8/2/44.

[24] Z. Xiao, Z. Li, Y. Liu et al., “Emrshare: a cross-organizational medical data sharing and management framework using permissioned blockchain,” in *Proceedings of the 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 998–1003, Singapore, December 2018.

[25] S. Jiang, J. Cao, J. Zhu, and Y. Cao, “Polychain: a generic blockchain as a service platform,” in *Proceedings of the International Conference on Blockchain and Trustworthy Systems*, pp. 459–472, Guangzhou, China, August 2021.

[26] Remix, “Remix online ethereum test network,” 2022, http://remix.ethereum.org.

[27] A. K. Shrestha, S. Joshi, and J. Vassileva, “Customer data sharing platform: a blockchain-based shopping cart,” in *Proceedings of the 2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pp. 1–3, Ontario, Canada, May 2020.

[28] P. Singh, M. Masud, M. S. Hossain, and A. Kaur, “Cross-domain secure data sharing using blockchain for industrial iot,” *Journal of Parallel and Distributed Computing*, vol. 156, pp. 176–184, 2021.

[29] A. Theodouli, S. Arakliotis, K. Moschou, K. Votis, and D. Tzovaras, “On the design of a blockchain-based system to facilitate healthcare data sharing,” in *Proceedings of the 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering*

(TrustCom/BigDataSE), pp. 1374–1379, IEEE, New York, NY, USA, August 2018.

[30] M. Divya and N. B. Biradar, "Iota-next generation block chain," *International Journal Of Engineering And Computer Science*, vol. 7, no. 04, pp. 23823–23826, 2018.

[31] M. Blum, P. Feldman, and S. Micali, "Non-interactive zero-knowledge and its applications," in *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, pp. 329–349, ACM, New York, NY, USA, 2019.

[32] L. Aumayr, O. Ersoy, E. Andreas et al., "Generalized channels from limited blockchain scripts and adaptor signatures," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*, pp. 635–664, Seoul, South Korea, December 2021.

[33] G. Malavolta, P. Moreno-Sanchez, A. Kate, M. Maffei, and S. Ravi, "Concurrency and privacy with payment-channel networks," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 455–471, Dallas, TX, USA, October 2017.

[34] G. Malavolta, P. Moreno-Sanchez, C. Schneidewind, A. Kate, and M. Maffei, "Anonymous multi-hop locks for blockchain scalability and interoperability," in *Proceedings of the 26th Annual Network and Distributed System Security Symposium, NDSS 2019*, California, CA, USA, February 2019.

[35] S. A. Thyagarajan, G. Malavolta, and P. Moreno-Sanchez, "Universal atomic swaps: secure exchange of coins across all blockchains," in *Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1299–1316, San Francisco, CA, USA, May 2022.

[36] H. Ren, H. Li, D. Liu, G. Xu, N. Cheng, and X. Shen, "Privacy-preserving efficient verifiable deep packet inspection for cloud-assisted middlebox," *IEEE Transactions on Cloud Computing*, vol. 10, no. 2, pp. 1052–1064, 2022.

[37] A. Yang, J. Xu, J. Weng, J. Zhou, and D. S. Wong, "Lightweight and privacy-preserving delegatable proofs of storage with data dynamics in cloud storage," *IEEE Transactions on Cloud Computing*, vol. 9, no. 1, pp. 212–225, 2021.

[38] M. Bellare and P. Rogaway, "Random oracles are practical: a paradigm for designing efficient protocols," in *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pp. 62–73, Virginia, VA, USA, November 1993.

[39] E. Andreas, S. Faust, K. Hostáková, M. Maitra, and S. Riahi, "Two-party adaptor signatures from identification schemes," in *Proceedings of the IACR International Conference on Public-Key Cryptography*, pp. 451–480, Daejeon, South Korea, May 2021.

[40] Eth gas station, "Eth gas station," 2022, https://ethgasstation.info.