symmetry

MDPI

Article

Data-Sharing System with Attribute-Based Encryption in Blockchain and Privacy Computing [†]

Hao Wu , Yu Liu *, Konglin Zhu and Lin Zhang

School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China; wuhaodoc@bupt.edu.cn (H.W.); klzhu@bupt.edu.cn (K.Z.); zhanglin@bupt.edu.cn (L.Z.)

* Correspondence: liuy@bupt.edu.cn

[†] This paper is an extended version of our paper published in Wu, H.; Wang, S.; Dong, G.; Zhou, M.; Liu, Y.; Zhang, L. Data Sharing System Based on Blockchain to Enhance The Visible of Meta Data Information in Privacy Computing. In Proceedings of the 2023 IEEE International Conference on High Performance Computing & Communications, Data Science & Systems, Smart City & Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), Melbourne, Australia, 13–15 December 2023; pp. 296–302, https://doi.org/10.1109/HPCC-DSS-SmartCity-DependSys60770.2023.00048.

Abstract: With the development of the data-sharing system in recent years, financial management systems and their privacy have sparked great interest. Existing financial data-sharing systems store metadata, which include a hash value and database index on the blockchain, and store high-capacity actual data in the center database. However, current data-sharing systems largely depend on centralized systems, which are susceptible to distributed denial-of-service (DDoS) attacks and present a centralized attack vector. Furthermore, storing data in a local center database has a high risk of information disclosure and tampering. In this paper, we propose the ChainMaker Privacy Computing (CPC) system, a new decentralized data-sharing system for secure financial data, to solve this problem. It provides a series of financial data information and a data structure rather than actual data on the blockchain to protect the privacy of data. We utilize a smart contract to establish a trusted platform for the local database to obtain encrypted data. We design a resource catalog to provide a trusted environment of data usage in the privacy computing system that is visible for members on the blockchain. Based on cipher-policy attribute-based encryption (CP-ABE), We design a CPC-CP-ABE algorithm to enable fine-grained access control through attribute based encryption. Finally, We propose an efficient scheme that allows authenticated data-sharing systems to perform Boolean searches on encrypted data information. The results of experiment show that the CPC system can finish trusted data sharing to all organizations on the blockchain.

Keywords: data-sharing system; blockchain; privacy computing; resources catalog; attribute-based encryption; Boolean search

check for updates

Citation: Wu, H.; Liu, Y.; Zhu, K.; Zhang, L. Data-Sharing System with Attribute-Based Encryption in Blockchain and Privacy Computing. *Symmetry* **2024**, *16*, 1550. https://doi.org/10.3390/sym16111550

Academic Editor: Tomohiro Inagaki

Received: 30 August 2024
Revised: 2 October 2024
Accepted: 3 October 2024
Published: 19 November 2024

1. Introduction

The rapid expansion of the Internet and the advent of big data have significantly intensified the focus on data, impacting entities ranging from government agencies and corporate sectors to individual users. As the scope of data utilization broadens, the full realization of data capacity and data value increasingly depends on seamless intercommunication, interoperability, and societal-wide integration. However, traditional data security frameworks are becoming increasingly insufficient in addressing the rising demands for robust security and compliance in data circulation. Ensuring the secure and orderly flow and utilization of data, while maximizing their inherent value, has become a critical challenge in the ongoing development of the digital economy. In recent years, blockchain and cryptographic primitives have emerged as two pivotal technologies in the realm of privacy protection, each sharing aligned goals but distinguished by their unique technical attributes.

Cryptographic primitives are a set of technology systems that includes artificial intelligence, cryptography, data science, and other fields rather than a single technology [1]. This technology can achieve encrypted or non-transparent data computation in the face of data owners, data collectors, data publishers, and data users, as well as attackers who intend to steal data. While providing full lifecycle protection for privacy information, It performs data computation and analysis, ensuring the 'availability and confidentiality' of the data. Blockchain technology is becoming increasingly popular in recent years [2]. It enables complete lifecycle management of data through technical features like traceability, tamper resistance, and automated smart contract execution. It ensures cross-verification of data authenticity before linking, and, once linked, the data become virtually tamper-proof and traceable. The blockchain is a sharp sword for solving multi-party collaboration and trust problems. The combination of the two can fully leverage their respective advantages, ensuring privacy protection in data sharing and providing feasible solutions for issues such as data authenticity and data authentication. The completion of the data-sharing process is recordable, verifiable, traceable, and auditable, laying a solid foundation for building an efficient, secure, and liquid data element market.

With the increasing concern of financial data privacy, many scholars have proposed many privacy-preserving financial sharing schemes [3,4]. In these schemes, data owners aggregate data from local databases and encrypt them applying defined access control policies before outsourcing data to cloud servers. Data users, characterized by diverse attributes, transmit Sending encrypted search terms to the cloud server to retrieve the encrypted data that they require. The cloud server then conducts search operations over the ciphertexts using these keywords. Users are only able to decrypt the ciphertext if their attributes satisfy the defined access control policies. However, this model relies on a centralized server to manage the system and process queries, which introduces two major drawbacks. First, the centralized structure is vulnerable to distributed denial-of-service (DDoS) attacks and centralized attack vectors, which could disrupt financial services. Second, the cloud server may not always be fully trustworthy and might fail to perform all required computations.

In this paper, we propose the CPC system, a new data system based on blockchain technology and cryptographic primitives technology for privacy-preserving financial data. To protect the security and privacy of financial data, the data owners design a resource catalog to record data structure information and data usage information before collecting actual data from the local database; then, they publish the encrypted data index on the blockchain via smart contracts, which then provide secure and reliable search services. For user security access to data, We propose a fine-grained on-chain access control mechanism using attribute-based encryption [5], embedding the access policy in the results. To optimize data retrieval, we design a non-interactive, sub-linear complexity Boolean search protocol on the chain based on the Boolean search scheme. This work has the following contributions:

- Combined with blockchain technology and privacy computing technology, we propose a data-sharing system in the blockchain network. We design a resource catalog to formulate data usage rules to data users, where data catalog registration means that the data provider will provide a metadata description, which is a comprehensive description of their data owner; note that this refers only to metadata, data description, and data structure, and not actual data or real business data. Calculation model catalog refers to the registration and publication of privacy computing models, which is the logic of calculation. Computing resource registration refers to the registration and publication of computing server resources, such as private computing, a trusted execution environment, and security multi-party computing.
- All data sharing is in blockchain networks. Compared with traditional access control, attribute-based encryption access control is more flexible and more secure. Attribute-based access control combines attribute sets to implement the data access. Blockchain technology ensures all data are recorded and traceable. Data owners have fine-grained,

one-to-many access control through a cipher-policy attribute-based encryption (CP-ABE) algorithm, and the design access policy, in our system, distributes private keys to data users, which eliminates direct interaction between data owners and users, reducing the data-sharing burden on data owners.

- We propose a novel privacy-enhancing on-chain Boolean search approach that facilitates efficient data retrieval before accessing plaintext information. This scheme allows a semi-trusted server to perform search operations directly over encrypted data without necessitating decryption, thereby ensuring robust protection of data privacy.

The rest of this paper is organized as follows: in Section 2, we review some related work and the background. In Section 3, we introduce preliminaries. In Section 4, we introduce the CPC system proposed in this paper. In Section 5, we discuss the security of the proposed system. In Section 6, we show the results of the performance test and comparison experiment. In Section 7, we conclude the paper and the future work.

2. Related Work

The requirement of an increase in data sharing is crucial to the process of a data-sharing system. Data-sharing systems and access control have been studied by many scholars. We summarize the work related to two aspects of this paper: the data-sharing system and access control.

Zheng et al. [6] proposed a secure and trusted data-sharing model for government data sharing. This model provides privacy protection and traceability. Ma et al. [7] proposed a data-sharing scheme based on the blockchain that relieves the storage pressure of the blockchain by storing encrypted data in a cloud database and transfers the ciphertext retrieval process to the blockchain to solve the untrustworthy problem of the cloud manager. Li et al. [8] proposed a data access control system in cloud blockchain integration that was designed and implemented using a decentralized, immutable blockchain and transparent, automated smart contracts to avoid the problems of centralized, traditional cloud services and non-transparent data logs. Medical data have high confidentiality and complexity. Dai et al. [9] proposed a (treatment data of diabetes mellitus type 2) T2DM data-sharing system. Huang et al. [10] proposed a privacy-preserving vehicular data-sharing framework based on the blockchain and designed an anonymous and auditable data-sharing scheme via zero-knowledge proof to protect the identity privacy of vehicles. Xu et al. [11] proposed a secure blockchain-based data trading system for vehicular crowd sensing to solve the problem of malicious behavior in data-sharing systems based on the blockchain. Yuan et al. [12] proposed a blockchain-based trusted data-sharing mechanism with congestion control in the Internet of vehicles to address the issue of extensive data sharing potentially causing channel congestion. Chen et al. [13] proposed a data-sharing privacy protection model to ensure the security and privacy of data and improve fairness and efficiency. Wang et al. [14] proposed an efficiently anonymous authentication scheme to ensure that a data accessor is authorized. Shen et al. [15] presented a block-design-based key agreement protocol for group data sharing in cloud computing, leveraging a symmetric balanced incomplete block design (SBIBD). The main goal is to address the challenges of security and efficiency in group data sharing in cloud environments where multiple participants are involved. Zhou et al. [16] proposed a framework for secure and trustworthy federated learning and data sharing in the Industrial Internet of Things (IIoT) using blockchain technology to ultimately aim toward improving privacy, security, and trust in federated learning and data sharing in the IIoT.

For access control in data sharing, many scholars have proposed an access policy scheme. Mei et al. [17] designed an unbounded and puncturable ciphertext-policy ABE with an arithmetic span program scheme and presented an expressive data-sharing and self-controlled fine-grained data deletion solution in cloud-assisted IoT. Han et al. [18] proposed a dual-strategy attribute-based encryption (ABE) scheme for distributed outsourcing and two access structures and a structure of attribute sets. Yan et al. [19] proposed a data access scheme based on attribute-based encryption in the blockchain environment. We employ an improved ciphertext-policy attribute-based encryption (CP-ABE) algorithm to provide fine-grained access to data under policy concealment. In summary, these functions solve the problem of privacy preservation and security, where the data users are unable to understand data information. For access control, these issues are still in their early stages and have board research prospects. Shen et al. [20] proposed a secure, traceable, and efficient method for accessing and sharing encrypted eHealth data, especially in emergency cases, while ensuring that unauthorized access is accountable. Rao et al. [21] proposed a novel secure searchable attribute-based signcryption (sABSC) scheme. This scheme is the first of its kind to integrate several important features: it enables Boolean formula-based searches over signcrypted data, ensures the privacy of keywords, allows for verifiable outsourced unsigncryption, and provides mechanisms for the self-verifiability of search results. Zhang et al. [22] proposed a blockchain (BC)-based anonymous attribute-based searchable encryption (ABSE) scheme for data sharing, referred to as BADS. The scheme is designed to enhance confidentiality by concealing the attributes within the access policy, thereby protecting the attributes that satisfy the policy. Huang et al. [23] proposed an attribute-based expressive and ranked keyword search scheme over encrypted documents, named ABERKS. This scheme allows authorized users to submit complex Boolean query formulas that include operators such as AND, OR, NOT, and threshold operators, enabling more expressive and precise searches over encrypted data. Scheme comparision is shown in Table 1.

Table 1. Scheme comparison.

Scheme	Data Leakage	Secure Attribute Management	Security Searchable Encryption
Scheme [11]	×	✓	✓
Scheme [13]	✓	✓	×
Scheme [15]	×	✓	✓
Scheme [18]	×	×	✓
Our CPC	✓	✓	✓

3. Preliminaries

3.1. Bilinear Pairing

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic groups of prime order p, with g being the generator of $\mathbb{G}$. Define $\hat{e}$ as a bilinear pairing, $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, satisfying the following properties:

1. Bilinear: $\hat{e}(g^a, g^b) = \hat{e}(g, g)^{ab}$ for all $a, b \in \mathbb{Z}_p^*$.
2. Non-degenerate: $\hat{e}(g, g) \neq 1$.
3. Computable: It is efficient to compute $\hat{e}(g_1, g_2)$ for all $g_1, g_2 \in \mathbb{G}$.

3.2. Hardness Assumptions

3.2.1. DDH Assumption

Let $\mathbb{G}$ be a cyclic group of prime order p. The decisional Diffie–Hellman (DDH) problem is to distinguish the ensembles $\{(g, g^a, g^b, g^{ab})\}$ from $\{(g, g^a, g^b, g^z)\}$, where the elements $g \in \mathbb{G}$ and $a, b, z \in \mathbb{Z}_p$ are selected uniformly at random. The DDH assumption holds if no probabilistic polynomial-time (PPT) distinguisher $\mathcal{D}$ can solve the DDH problem with a non-negligible advantage.

$$\Pr[\mathcal{D}(\mathbb{G}, p, g, g^a, g^b, g^z) = 1] - \Pr[\mathcal{D}(\mathbb{G}, p, g, g^a, g^b, g^{ab}) = 1] < \epsilon.$$

3.2.2. Strong RSA Problem

Let $n = pq$, where p and q are large primes. A random element is chosen from $\mathbb{Z}_n^*$. An algorithm $\mathcal{A}$ is said to solve the strong RSA problem if, given the input tuple (n, g), it outputs two elements (z, e) such that $z^e = g \mod n$.

3.3. Pseudo-Random Functions

A function F maps an element $x \in X$ to an output $y \in Y$ using a secret key $k_f \in K$. We say F is a pseudo-random function (PRF) if, for all efficient adversaries $\mathcal{A}$, its advantage satisfies

$$\mathrm{Adv}^{\mathrm{prf}}_{F,\mathcal{A}}(\kappa) = \left|\Pr[\mathcal{A}^{F(k,\cdot)}(1^{\kappa}) = 1] - \Pr[\mathcal{A}^{f(\cdot)}(1^{\kappa}) = 1]\right| < \mathrm{negl}(\kappa),$$

where $\mathrm{negl}(\kappa)$ is a negligible function and f is a truly random function from X to Y.

3.4. Access Policy in Attribute-Based Encryption

Access Structure Given an attribute set L and an access policy $\mathcal{AP}$, we say that L satisfies $\mathcal{AP}$ if $\mathcal{AP}$ returns 1 on L, denoted as $L \models \mathcal{AP}$. Otherwise, we write $L \not\models \mathcal{AP}$. In our scheme, we consider the AND-gate policy AND^*_m. Specifically, for an attribute list $L = [L_1, L_2, \ldots, L_n]$ and an access policy $\mathcal{AP} = [\mathcal{AP}1, \mathcal{AP}2, \ldots, \mathcal{AP}n] = \bigwedge i \in I\mathcal{AP}\mathcal{AP}i$, where $I\mathcal{AP}$ is the index set and $I\mathcal{P}_i = i | 1 \leq i \leq n, \mathcal{AP}_i \neq *$, $L \models \mathcal{AP}$ if $\mathcal{AP}_i = *$ or $L_i = \mathcal{AP}_i$ for all $1 \leq i \leq n$, and $L \not\models \mathcal{AP}$ otherwise. The wildcard $*$ in $\mathcal{AP}$ denotes a "do not care" value.

4. Our Proposed Mechanism

4.1. Threat Model

In our threat model, we assume the blockchain is trusted, while other organizations are honest but curious. They follow the transaction rules strictly but are interested in accessing sensitive data. Our main goal is to protect data confidentiality and query privacy in data access by the blockchain-based system. Similar to [2], we introduce the threat model in the following:

- In a data-sharing system, attackers may intercept and analyze transactions or data exchanges within the blockchain network to infer sensitive information. Even though data are encrypted, adversaries can attempt to deduce valuable information by observing data distribution patterns, metadata, or query requests. Data leakage attacks pose a significant threat to user privacy, particularly in decentralized data sharing environments.
- Since encryption relies heavily on the security of key management, any vulnerability in the key distribution, storage, or update process can allow attackers to bypass encryption protections and access sensitive data. Key leakage can lead to the compromise of a large volume of data, making robust key management strategies critical for ensuring data security.
- In attribute-based encryption (ABE) systems, users may collude by sharing their decryption keys to collectively gain unauthorized access to encrypted data. Collusion attacks enable multiple users to bypass access control policies by combining their attributes to decrypt sensitive information. This is a significant risk in systems where access control is implemented through attribute-based encryption.

4.2. System Model

Based on blockchain technology and privacy computing technology, we propose a ChainMaker Privacy Computing Sharing (CPC) system. The system consists of many ChainMaker Privacy Computing Platforms (CPCPs) and the blockchain in the system. One CPCP represents a system named management domain in one organization. Each organization defines the rules of data usage, including data computation and data storage. The organization becomes a trusted node through blockchain certificate authentication, becoming a member in the blockchain. The uncertified organization is unable to participate in data sharing. For the process of data sharing, one organization creates a resource catalog to record meta information and computation model information, and defines access control policies for data to determine which data users can obtain data in the CPC system. The system is shown in Figure 1.

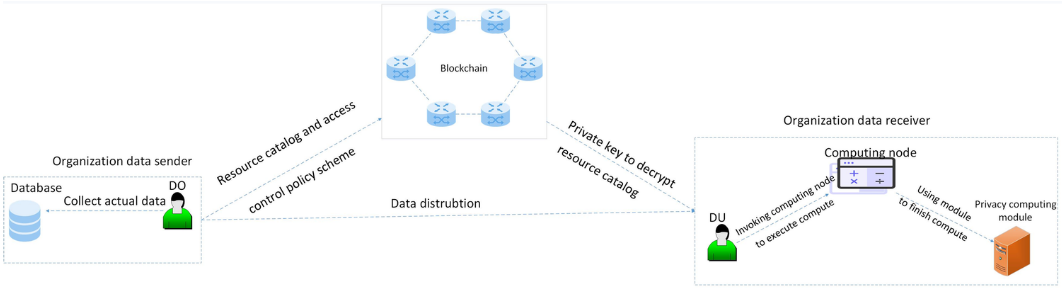


Figure 1. CPC system framework.

There are two main components in the CPC system.

- Organization management domain: The organization represents a member in the blockchain network, such as a company, enterprise, or department. Organizations develop a set of data usage rules and data access policies, and are able to collect actual data, transfer data, and execute data.
- Blockchain: We choose a permission blockchain where anyone can participate and access anonymously. The blockchain consists of a series of linked data blocks, added through consensus among peer nodes. Each link between blocks is created using cryptographic hash functions, ensuring the immutability of transaction data and the integrity of the block chain. Smart contracts are executed autonomously based on predefined logic, eliminating the need for a central authority, with the outcomes securely recorded on the blockchain.

There are two organization management domains in the CPC system. There is a CPCP in each organization. In the CPCP, the data owner uploads data information to the blockchain via invoking a smart contract to issue a "publish data to blockchain" transaction, and designs a computation model catalog to show the environment of the computing resource, such as a federated learning environment, trusted execution environment, and security multi-party computing. The data user in other organizations sends the request of the query data via a query smart contract to issue an "obtain data from blockchain" transaction. The CPC verifies the identify of the data user by a smart contract to determine whether the data user has permission to access.

There are five entities in the CPCP:

- Data Owners (DOs): In one organization, DOs collect actual data in the local database and provide the environment of computation via a resource catalog. They convert actual data into metadata and provide the the best privacy computing strategy based on data features and usage scenarios in the resource catalog.
- Data Users (DUs): In other organizations, DUs query-encrypted data information from the data catalog is published on the blockchain. When DUs access the resource catalog, they convert the search keywords into a search token using the authorized keyword key independently. The smart contract needs to verify whether the user's attributes satisfy the attribute values specified in the access control list.
- Computing Node: The network nodes participating in privacy computing can be located in different management domains in the business flow, and can represent software, computers, virtual computers, or clusters.
- Privacy Computing Module: There are three main modules in the CPCP, consisting of multi-party computing, federated learning, and a trusted execution environment. This module performs calculation and analysis on encrypted data or in an opaque state. Using privacy computing models to compute actual data guarantees the data privacy and security.
- Database: The database stores a number of actual data. This module is a precondition for data sharing and data distribution.

When DOs want to share data with DUs in the CPC system, they first collect actual data from the local database and provide a resource catalog. Then, DOs need to encrypt data via the CP-ABE algorithm and construct searchable indexes on the blockchain. DUs send the request to the blockchain with the attribute, the smart contract maps the attribute in the policy, DUs obtain the encrypted data, and then they choose the privacy computing algorithm that is recommended by the resource catalog. For instance, if DUs want to obtain the computed result, they need to obtain the computation model to finish computing via federated learning. DUs obtain encrypted data from DOs and obtain a private key to decrypt data. The specific data-sharing process is shown in Figure 2:

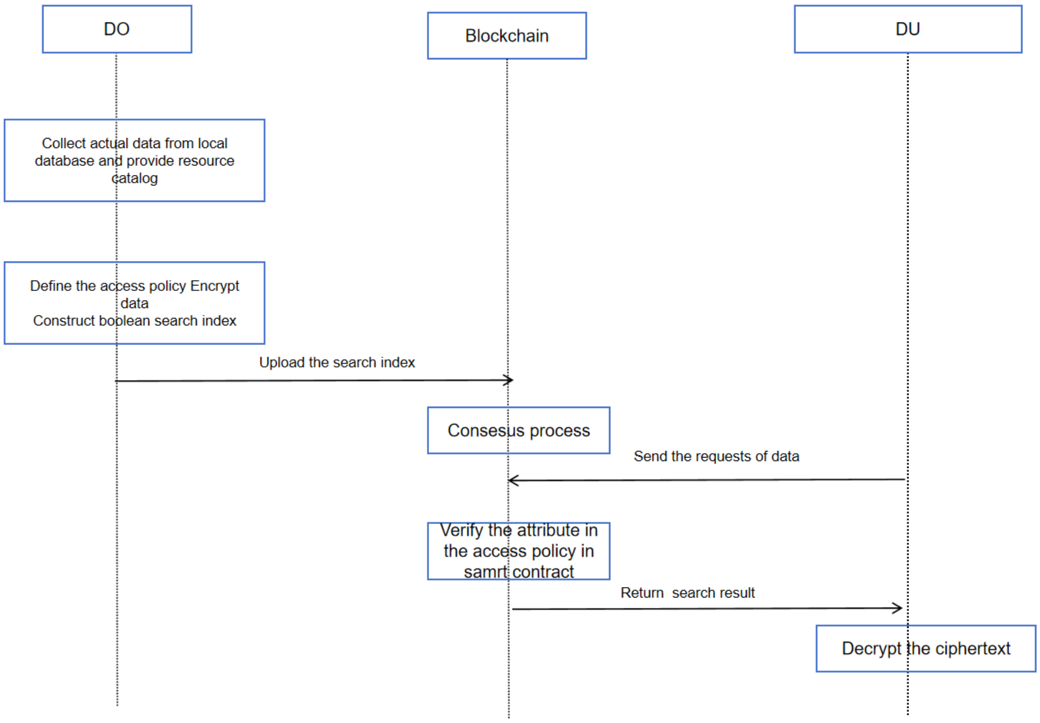

DO
Blockchain
DU
Collect actual data from local database and provide resource catalog
Define the access policy Encrypt data
Construct boolean search index
Upload the search index
Consesus process
Send the requests of data
Verify the attribute in the access policy in samrt contract
Return search result
Decrypt the ciphertext

Figure 2. The process flow of data-sharing system.

(1) Resource catalog storage: In each organization management domain, DOs collect actual data from the local database, design a data catalog to record the information of metadata, adopt the method of HANDLE to identify the data source to query it for DUs, provide the environment of computing and the define access control structure to upload to the access control list in the smart contract, and need to send a transaction request to blockchain networks. The blockchain verifies this transaction request and generate new blocks. Once new blocks are created, the resource catalog stores in the blockchain via invoking a storage smart contract.

(2) Resource catalog query: For other organization management domains, DUs send the requests of obtaining a resource catalog. The blockchain issues certificate authority (CA) to organization nodes to verify their identity and to establish secure communication. When the organization is trusted nodes, organization nodes invoke a query smart contract to obtain the resource catalog.

(3) Resource catalog access permission: When DUs send requests to the blockchain, the blockchain records the attribute of the DU, invokes an attribute management smart contract mapping policy in the access control list with the attributes of the DU, and uses digital signature technology or zero knowledge proof to implement access control.

(4) Data distribution: DUs send the requests of the data and computing model. When the DUs satisfy the policy of access control, they need to gain the identification of DOs from data owner organization in data catalog information and send the obtain data request. When the DU uses a federated learning algorithm, the executing computation party provides a trained model to the result party, the data user uses a trusted environment

extension, the data party provides homophobic encryption to the computing party, and then the computing party computes the encrypted result to the result party.

4.3. Catalog Structure and Management

Privacy computing is built by three parties: the data party provides data, the computing party provides computing sources, and the result party receives the results of computing. Data are encrypted in the computing process and are made invisible to the result party. So, we design a resource catalog to address the problems.

A resource catalog includes a data catalog and computing catalog. Each organization takes the responsibility of updating the data catalog based on the change in data. The data owner in CPC has the function of updating, uploading, and deleting the resource catalog. All data in a data-sharing system are maintained on a unified organization. Data in different organizations may be distinct from each other. Resource catalog information consists of data usage information, including data catalog information and computing model catalog information. When the DO uploads the resource catalog to the blockchain, the resource catalog has been authenticated through blockchain authorization and published on the blockchain network to show all organizations. Other organizations obtain metadata information from the blockchain via invoking a query smart contract.

To realize the efficient access control of the resource catalog in a CPC system, we classify all data into four levels based on some rules and data attributes. Among these four levels, the requirement to DUs of first-level data is the weakest while the requirement of fourth-level data is the most strict. Different levels of data have different requirements of data users' attributes. Only if the attributes of the DU satisfy the requirement of target data resources can the access control request be passed. DUs have a number of attributes; the DO makes decisions on the access to the private data based on security access policies. In our data-sharing access control framework, resource catalog information is on the blockchain. It consists of data catalog information, computing model information, authorization policy information, and operation records information. Among the resource catalog, the DO designs the structure of data usage access permission to determine whether the data user obtains the resource catalog. The data catalog and computation model catalog are as follows:

- Data catalog information: The data owner inverts actual data into metadata to provide data catalog information. When DOs use data in a computing model from the data catalog, executing the computing process needs to be mapped onto the actual data. Data catalog information consists of data classification information, data field information, and data access permission information. A standard catalog pattern is used. Before a data catalog is recorded on the blockchain, the DOs have to fill the data information as the standard catalog pattern. Table 2 describes the standard data catalog form on the blockchain as follows. We use 18 classes to describe data information in detail. All of these classes are helpful for data users searching for their needed source data. In real-world applications, dictionaries in a dictionary is a possible data structure for maintaining massive attributes.
- Computation model information: Besides the data catalog information, computing model catalog information consists of an algorithm model, disk resource, the environment of computing, the data field of model usage, and the model description. The algorithm model includes federated learning, security multi-party computing, privacy set interaction, and a trusted environment extension. The data field of model usage refers to the usage function of actual data, such as multi-party average and multi-party aggregation. The model description represents the algorithm model of the usage algorithm model. The computing party uses privacy set interaction to choose if they need data, and utilizes a proper algorithm model to finish the computing process. The computing results are saved in the CPC system.

Table 2. Data catalog description.

Data Name	Data Abstract	Data Starting Time
Data Updating Time	Data Format	Field Names
Data Type and Length	Major Key (Yes/No)	is Null (Yes/No)
Field Description	Value Range	Data Examples
Sharing Type	Sharing Condition	Data Size
Data Owner Organization	Data Level	Other Comments

Authorization policy information consists of all authorization policy management information, which is provided by data owners. All these authorization policies are put into smart contracts. Operation record information consists of all operation and action records, through which all nodes on the blockchain can realize the data usage tracing on the blockchain.

4.4. CP-ABE Algorithm

The CPC system utilizes CP-ABE to design its access control mechanism. To avoid relying on trusted third parties and to minimize the direct interaction between data owners and users, we implement most of the access control functionalities through smart contracts. This approach ensures automated enforcement of access policies while maintaining decentralization and security.

(1) Setup: The setup function is implemented in the CPC system and is mainly used to obtain the authority center's system master key and public key associated with symmetric key encryption. Each authority collects the attribute of the system and the attribute of the user.

In Algorithm 1, the global public parameters, GPs, are N and a generator $g1$ of G_{p1}. In addition, the description of a hash function

$$H : \{0,1\}^* \to G$$

that maps global identities, GIDs, to elements of G is published. In our proof, we will model H as a random oracle.

```
Algorithm 1: Setup
  Input: security parameter $\lambda$
  Output: Secret Key, Public Key and Authority Key
  Choose a bilinear group $\mathbb{G}_0$ of prime order p with generator g, and a bilinear map
    e: $\mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_1$.
  Choose two random exponents $\alpha, \beta \in \mathbb{Z}_p$ , and obtain $h = g^{\beta}$.
  Generate the system master key: SK = $(\beta, g^{\alpha})$
  Generate the public key: PK = $(\mathbb{G}_0, g, h, e(g,g)^{\alpha})$
```

(2) Encrypt: To ensure the security and privacy of data information, the data owner needs to encrypt the data information with their own generated symmetric key in their own organization and then upload the encrypted data information and symmetric key with data users to the blockchain. The detail is shown in Algorithm 2.

```
Algorithm 2: Encryption
    Input: Plaintext message M, access control(R,$\rho$)
    Output: The ciphertext message CT
    Choose a random s $\in \mathbb{Z}_N$ and choose a random v $\in \mathbb{Z}_N^l$ with s as its first entry.
    Let $\ngeqq_x$ denote $\mathbb{A}_x \cdot v$, where $\mathbb{A}_x$ is row x of A.
    Choose a random vector w $\in \mathbb{Z}_N^l$ with 0 as its first entry.
    Let $\supsetneqq_x$ denote $\mathbb{A}_x \cdot w$, for each row $\mathbb{A}_x$ of A, it chooses a random $\mathbb{R}_x \in \mathbb{Z}_N^l$.
    The ciphertext is computed as: $C_0 = Me(g1, g1)^s, C_1 = e(g1, g1)^{\lambda_x} e(g1, g1)^{\alpha_{\rho(x)} r_x}, C_{2,x} = g_1^{r_x}, C_{3,x} = g_1^{y_{\rho(x)} r_x} g_1^{\omega_x} \forall x$
```

(3) KeyGen: Compared with traditional encryption, attribute-based encryption is more flexible: it mainly consists of a set of attribute sets private from the attributes of data users. The key generation function is the key to "one-to-many" access control. Access control policy is a logical expression based on attribute sets that includes an "and", "or", "nor" three logical symbol. To ensure the security of the private attribute key, after the private key generated, it is distributed from a secure channel in the blockchain. The detail is shown in Algorithm 3.

```
Algorithm 3: KenGeneration
    Input: Global identities GID, attribute i, Sercet Key SK
    Output: Authority key $K_{i,GID}$
    Create a key for GID for attribute i belonging to an authority, the authority
      computes: $K_{i,GID} = g_1^{\alpha_i} \cdot H(GID)_i^y$ H is a random oracle.
```

(4) Decrypt: The data user obtains the private attribute key, which consists of the attributes of the data user. When the private key maps access control policy and satisfies a logical relationship, the private key decrypts the encrypted data and enables access to the original data.

In Algorithm 4, the decryptor has the secret keys $K_{\rho(x)}$ for a subset of rows AP_x of AP such that (1, 0, . . . , 0) is in the span of these rows. For each such x, the decryptor computes the following: $C_{1,x} \cdot e(H(GID), C_{3,x}) / e(K_{\rho(x),GID}, C_{2,x})$=$e(g1, g1)^s$.

```
Algorithm 4: Decryption
    Input: Ciphertext message CT, authority key $K_{i,GID}$
    Output: Plaintext message M
    Assume the ciphertext CT is encrypted under an access matrix (AP, $\rho$).
    The decryptor then chooses constants $c_x \in \mathbb{Z}_N$ such that $\sum_x c_x \cdot AP = (1, 0, ..., 0)$
      and computes:
          $$\prod_x \Big( e(g1, g1)^{\lambda_x} \cdot e(H(GID), g1)^{\omega_x} \Big)^{c_x} = e(g1, g1)^s$$
    The message can then be obtained as: $M = C_0 / e(g1, g1)^s$
```

4.5. Boolean Search Algorithm

To facilitate presentation, we will denote the corresponding prime of the keyword as w. Subsequently, leveraging the RSA function, data providers need to interact only once to issue keyword secret keys during the initialization phase. This design allows users to independently generate search tokens for their desired keywords, thereby significantly reducing the communication overhead between the DO and DU. Our scheme guarantees that users can conduct secure keyword searches exclusively within the authorized keyword set.

Given the prime values for the keywords, the Data Owner (DO) uses $g^{\frac{1}{w}}$ to create secure searchable indexes for the keyword set w_i. To authorize access to the keyword set, each DO provides potential users with a partial token $g^{\frac{1}{w_1 \cdots w_n}}$ for the set $w_1, w_2, \ldots, w_n$, which represents the keywords users can search. As a result, a user's search capabilities are restricted to a specified range, rather than allowing searches over random keywords. When a user searches for a keyword $w_i \in \mathbf{w}$, they can retrieve the value $g^{\frac{1}{w}}$ from Equation (1) based on the strong RSA problem, which is then used to generate the encrypted search tokens

$$g^{\frac{1}{w_i}} = \left(g^{\frac{1}{w_1 \cdots w_n}} \right) \prod_{w \in \mathbf{w} \setminus \{w_i\}} w. \tag{1}$$

Building on the previous design, our focus shifts to enabling encrypted Boolean search within a blockchain-based data-sharing framework. A simple approach to implementing Boolean search involves conducting individual keyword searches repeatedly to retrieve all matching files, and then having the data user (DU) intersect the results to identify the common files. This method necessitates multiple interactions between the server and users, increasing communication overhead. Additionally, it poses significant privacy risks, as the server gains insight into the matched files for each keyword. To address these issues, we propose an efficient and secure searchable encryption scheme tailored for Boolean search on the blockchain.

We aim to construct three on-chain indexes to achieve sublinear search complexity within the blockchain. These indexes are represented as key-value pairs:

- **EDindex** (encrypted DB index) is an encrypted mapping from keywords to all corresponding file IDs.
- **BSindex** (Boolean search index) is a file-keyword mapping where the key denotes the file's association with the keyword, and the value indicates whether this association exists.
- **PTindex** (partial token map index) is used to generate search tokens (as discussed in Section 6.1), eliminating the need for a trusted server as detailed in [16].

To ensure the privacy of search keywords, we begin by encrypting them into $stag_w$ (as detailed in *IndexGen* in Section 6.1). We then compute $zind \leftarrow F_p(K, t, id)$ to obscure the file ID id using the pseudorandom function (PRF) F_p. As illustrated in Figure 3, within the EDindex, each file ID is linked to a counter c and a random nonce t_c, with each keyword w corresponding to multiple files in the inverted database index. To ensure uniqueness for each file ID, we generate the search token nonce st_c by applying a pseudorandom permutation (PRP) P to the random nonce t_c of the current file ID and the previous st_{c-1}. This guarantees that the connection between newly added files and previous search queries remains concealed. For the key generation in EDindex, each keyword is associated with the latest random token nonce st_c using hash function H_1. Additionally, another distinct hash function, H_2, is employed to mask both the ciphertexts and the random nonce t_c associated with the file ID as follow:

$$e \leftarrow (e_{id} || t_c) \oplus H_2(stag_w || st_c), \tag{2}$$

Here, e_{id} represents the encrypted ciphertext of ABE as defined in Equation (3), which enables fine-grained on-chain access control. The access control policy is then securely embedded in the result on the blockchain within the MedShare framework. Furthermore, to link each keyword w to the corresponding file counter c (for the multi-keyword Boolean search), we generate a blinding value z:

$$z \leftarrow F_p\left(K_z, g^{1/||c||}\right). \tag{3}$$

Next, we compute the blinded value $y = zind \cdot x^{-1}$, which is used to generate the search token after performing the Boolean search. In BSindex, to facilitate Boolean search, we establish and pre-store a one-to-one mapping that indicates the presence of a keyword in a file. This is done by calculating $ztag = g^{F_p(K_x,y) \cdot zind} \cdot z$. The resulting $stag$ value is then used to verify if the file ($zind$) contains the keyword w'.

From the search process, multiple keywords "$w_1, w_2, \ldots$" are included in query Q. where w_1 is the least frequent keyword. The blinding value z s then used to blind the remaining search keywords w_j ($j = 2, \ldots$) in order to construct the $ztoken[i,j]$ as follows:

$$g^{F_p(K_z(s_k)) \prod_{w \in \mathbf{w} \setminus \{w_j\}} w} = g^{F_p(K_z s_k) \prod_{w \in \mathbf{w} \setminus \{w_j\}} w}, \tag{4}$$

where i denotes the counter for the ith file containing the keyword w_1, and $s_k^{(2)}, s_k^{(3)}$ are components of the search authorization key (as specified in Equation (4)). To prevent interaction between the server and the index, PTindex is designed to associate the counter c and search token nonce st_c with the current keyword. Here, Only users with valid search keys can generate the label l to retrieve the designated partial token from PTindex via smart contracts. The file ID is added to the final result only when all $ztoken[i,j] \in$ BSindex for the ith file of w_1 will this id be added to the final result. This protocol is correct because $ztoken[i,j] = g^{F_p(K_x s_j y)} \cdot z^{\left(zind \cdot x^{-1}\right)}$, which means that this $ztag$ in BSindex is correctly recomputed. If all $ztags$ are matched in BSindex, this file also contains all the rest keywords in query Q.

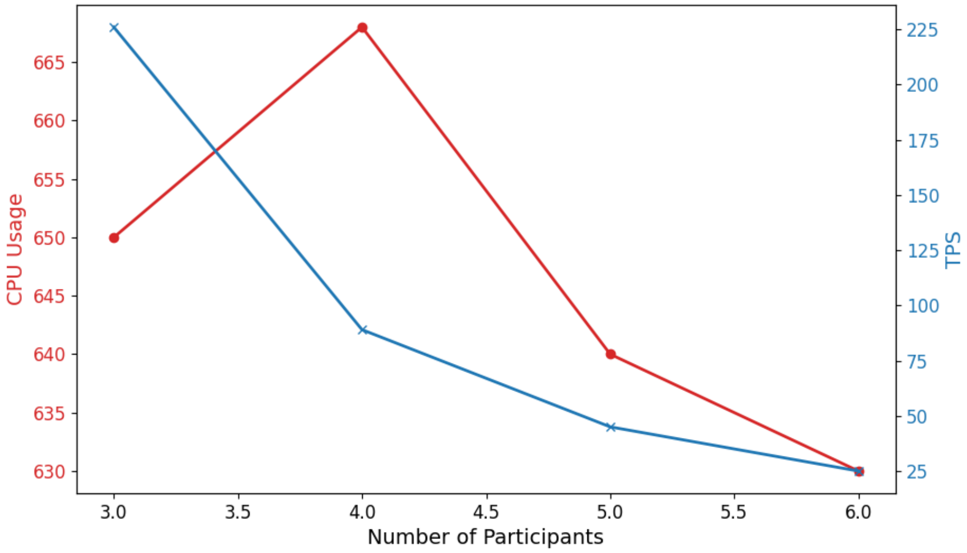


Figure 3. Evaluation for CPC performance.

5. Security Analysis

The CPC system that is proposed in this paper combines the blockchain and attribute-based encryption, and designs permission blockchain technology to provide better security compared to a data-sharing system in the public blockchain. This chapter examines system security grounded in cryptographic assumptions and blockchain security mechanisms.

5.1. Security of Data Sharing

Our system, attribute-based encryption (ABE), ensures that data remain encrypted during transmission and storage. Since the data are stored on the blockchain, the immutable and traceable nature of the blockchain enhances data security. To prevent data leakage, our system ensures privacy protection for query requests and metadata. Implementing unlinkable attribute sets or utilizing zero-knowledge proof (ZKP) could guarantee that even if attackers gain access to metadata or query patterns, they cannot infer sensitive information. DOs collect actual data that do not leave the local organization; they only show metadata information and computing model description to others on the blockchain. This ensures the security of the CPC system.

5.2. Security of CPC-CP-ABE

The CP-ABE algorithm is designed based on cipher-policy attribute-based encryption to prevent collusion attack; it uses the "ties attributes" function, which is various global attributes belonging to specific users, to guarantee the security of the CPC system. When attackers decrypt the ciphertext, they must recover the blinding factor $e(g1, g1)^s$ via their attribute to map their global identity, GID. If the user has a set of attribute keys with the same global identity, GID, these additional terms will cancel in the decrypt algorithm. If attackers have two different global identities, GID and GID′, it means that there are two different attributes: the result is e(g1,H(GID)) and e(g1,H(GID′)), which will not cancel each other out.

Lewko et al. [24] proved the security of the CP-ABE algorithm based on the above assumption. Attribute-based encryption mainly prevents attacks by collusion. Once the attributes of one entity, rather than many attributes, combine, the user decrypts the plaintext message.

5.3. Security of Result Storage

To further protect against key leakage, techniques like secure multi-party computation (SMPC) or threshold encryption can be employed, ensuring that only authorized parties can decrypt data through collaboration. There is a chain structure in the blockchain, where each block connects the previous block; once a block is modified, then all blocks are altered, and the function of being tamper-proof is possessed to guarantee security. DUs send the computing process via data catalog information and computing model catalog information, and the blockchain approves these requests; this process becomes a trusted process. The result of computing the coverage intermediate encrypted state is from multi-party calculations and is to be stored in local organizations; this result is shared with DUs who send the request. This process is carried out to ensure the security and privacy of data.

6. Experiment

6.1. Experimental Settings

We evaluate the performance of a data-sharing system based on a permission blockchain system built on the ChainMaker platform. ChainMaker is a permission blockchain where anyone can join and participate anonymously, and participants in a permission blockchain are usually known and verified. This is often carried out through a trusted authority. The permission blockchain has mechanisms to control who can join the network, who can view the blockchain, and who can participate in maintaining the blockchain. In secure multi-party computation, we use an oblivious transfer protocol to show the result of two-party computation and an oblivious replicated secret sharing protocol to show the result of three-party computation. These protocols are used to protect privacy and security during the computation process. In three-party computation, each party has 10,000 floating-point data, and the three-party data are multiplied to obtain the final calculation data. In the process of data sharing, we test the times of the computation and the memory usage of two nodes and calculate TPS in three-party computation by Formula (5), which represents the amount of data processed within a single-core system in one second during computational tasks. The operating system of the testing machine is based on Ubuntu v20.04LTS, with 16 GB RAM and 64 logic CPU cores.

$$TPS = \frac{\text{Data volume} \times 2 \times 64 \times 100}{\text{times} \times \text{CPU usage} \times 3} \tag{5}$$

6.2. Result

In order to assess the operational efficiency of the CPC system, we first conducted tests on the data-sharing time across different numbers of organizations. Specifically, we recorded the end-to-end data-sharing time when the number of organizations was two to six. The data-sharing time includes the time taken to provide data, calculate, and save the results. We implement the TPS and CPU usage in Figure 3. The data storage time and data query time are determined by the creation of a new blockchain through ChainMaker. Therefore, we present the computation times for scenarios involving two parties and three parties in Tables 3 and 4. The results demonstrate that the CPC system completes data sharing securely and accurately.

Table 3. Two-party performance calculation.

	Data Volume (w)	Times (ns)	CPU Usage	TPS
integer multiplication	1	9.929	428	1505.91
integer comparison	1	8.044	386	2061.01
floating-point multiplication	1	10.852	412	1434.36
floating-point comparison	1	10.495	390	1563.49

Table 4. Three-party performance calculation.

	Data Volume (w)	Times (ns)	CPU Usage	TPS
integer multiplication	1	18.741	200.6	2269.76
integer comparison	1	32.082	200.6	1325.94
floating-point multiplication	1	19.623	208	2090.67
floating-point comparison	1	35.505	208.6	1152.43

Then, we evaluated the efficiency of attribute-based encryption. We compared the performance of the CPC system on a smart contract to that of the CP-ABE tool on bare metal. Consistent with theoretical expectations, the time for generating attribute keys increases with the number of user attributes. We demonstrated the feasibility of CP-ABE running time in the CPC system from Figure 4. We show encryption time and decryption time in Figure 5.

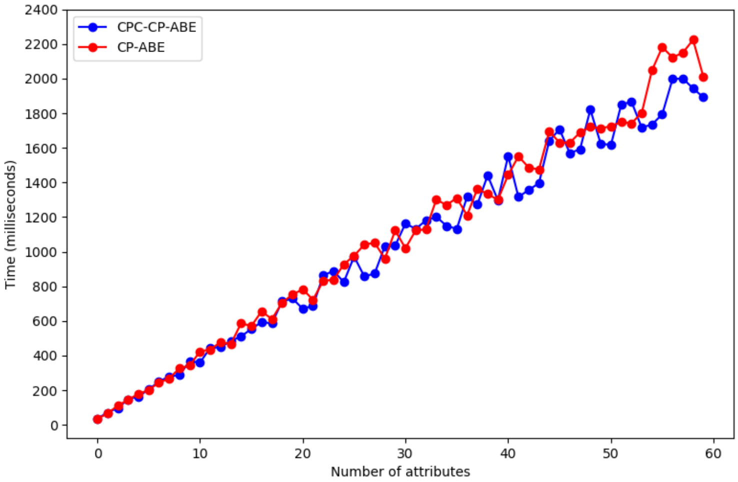
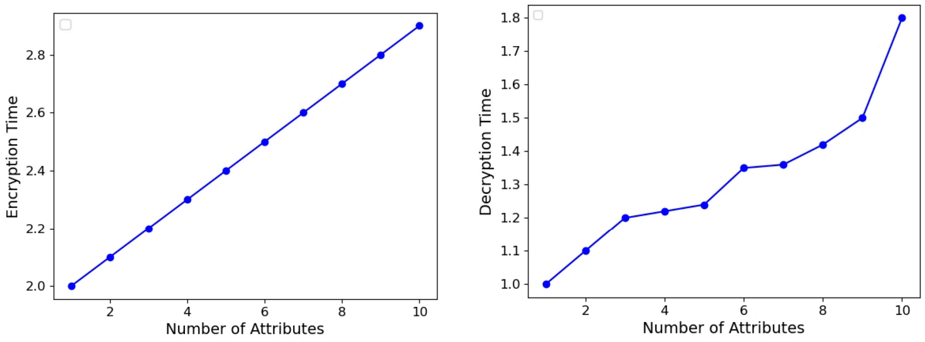


Figure 4. Comparison of attribute key generation time between the CPC on a smart contract and CP-ABE on bare metal.



(**a**) (**b**)

Figure 5. Comparison of the computation time. (**a**) Encryption; (**b**) decryption.

7. Conclusions

In this paper, we propose a data-sharing system based on blockchain technology and privacy computing technology, taking into account their traceability, auditability, and tamper-proof nature. We design a resource catalog to provide metadata and computational model information to the data user. The data owner provides a trusted data source, while the executing computation party obtains the actual data to complete the computing process and records the result in the local organization. The data user then collects the data result from the computing party. This process not only ensures data security and privacy but also implements visibility of data structure and data information in privacy computing. We further design a CPC-CP-ABE algorithm to encrypt the resource catalog and design an access control structure, enabling data owners to perform fine-grained access control over the data. Data users can meet the requirements by decrypting the resource catalog. In future work, we plan to expand the capabilities of our proposed data-sharing system by addressing scalability and performance optimization as the number of participating organizations increases.

Author Contributions: Conceptualization, H.W., Y.L. and K.Z.; methodology, H.W., Y.L. and L.Z.; software, H.W.; validation, Y.L., K.Z. and L.Z.; formal analysis, H.W.; investigation, K.Z.; resources, K.Z. and L.Z.; data curation, Y.L., K.Z. and L.Z.; writing—original draft preparation, H.W.; writing—review and editing, Y.L., K.Z. and L.Z.; visualization, H.W.; supervision, Y.L.; project administration, L.Z.; funding acquisition, K.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by National Key Research and Development Program of China (2023YFB2704500), Beijing Natural Science Foundation (4222033) and Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Epiphaniou, G.; Pillai, P.; Bottarelli, M.; Al-Khateeb, H.; Hammoudesh, M.; Maple, C. Electronic Regulation of Data Sharing and Processing Using Smart Ledger Technologies for Supply-Chain Security. *IEEE Trans. Eng. Manag.* **2020**, *67*, 1059–1073. [CrossRef]
2. Wang, Z.; Chen, Q.; Liu, L. Permissioned Blockchain-Based Secure and Privacy-Preserving Data Sharing Protocol. *IEEE Internet Things J.* **2023**, *10*, 10698–10707. [CrossRef]
3. Li, H.; Yang, Y.; Dai, Y.; Bai, J.; Yu, S.; Xiang, Y. Achieving Secure and Efficient Dynamic Searchable Symmetric Encryption over Medical Cloud Data. *IEEE Trans. Cloud Comput.* **2017**, *8*, 484–494. [CrossRef]
4. Xu, L.; Sun, S.; Yuan, X.; Liu, J.K.; Zuo, C.; Xu, C. Enabling Authorized Encrypted Search for Multi-Authority Medical Databases. *IEEE Trans. Emerg. Top. Comput.* **2019**, *9*, 534–546. [CrossRef]
5. Wang, M.; Guo, Y.; Zhang, C.; Wang, C.; Huang, H.; Jia, X. MedShare: A Privacy-Preserving Medical Data Sharing System by Using Blockchain. *IEEE Trans. Serv. Comput.* **2021**, *16*, 438–451. [CrossRef]
6. Zheng, Q.; Guo, B.; Hu, Y.; Li, Z. A Secure and Trusted Data Sharing Scheme Based on Blockchain for Government Data. In Proceedings of the 2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science

& Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), Hainan, China, 18–20 December 2022; pp. 936–942. [CrossRef]

7. Ma, X.; Wang, C.; Wang, L. The Data Sharing Scheme based on Blockchain. In Proceedings of the 2nd ACM International Symposium on Blockchain and Secure Critical Infrastructure (BSCI '20), Taipei, Taiwan, 6 October 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 96–105. [CrossRef]
8. Li, S.; Li, R.; Zhang, Y.; Huang, Y. CBI: A Data Access Control System Based on Cloud and Blockchain Integration. In Proceedings of the 2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Yanuca Island, Cuvu, Fiji, 14–16 December 2020; pp. 715–721. [CrossRef]
9. Dai, W.; Lu, Z.; Xie, X.; Wang, D.; Jin, H. Diabetes Mellitus Type 2 Data Sharing System Based on Blockchain and Attribute-Encryption. In Proceedings of the 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Wuhan, China, 9–11 December 2022; pp. 498–505. [CrossRef]
10. Huang, J.; Kong, L.; Wang, J.; Chen, G.; Gao, J.; Huang, G.; Khan, M.K. Secure Data Sharing over Vehicular Networks Based on Multi-sharding Blockchain. *ACM Trans. Sens. Netw.* **2024**, *20*, 1–23. [CrossRef]
11. Xu, H.; Qi, S.; Qi, Y.; Wei, W.; Xiong, N. Secure and Lightweight Blockchain-based Truthful Data Trading for Real-Time Vehicular Crowdsensing. *ACM Trans. Embed. Comput. Syst.* **2024**, *23*, 1–31. [CrossRef]
12. Yuan, M.; Xu, Y.; Zhang, C.; Tan, Y.; Wang, Y.; Ren, J.; Zhang, Y. TRUCON: Blockchain-Based Trusted Data Sharing With Congestion Control in Internet of Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 3489–3500. [CrossRef]
13. Chen, Y.; Li, J.; Wang, F.; Yue, K.; Li, Y.; Xing, B.; Zhang, L.; Chen, L. DS2PM: A Data-Sharing Privacy Protection Model Based on Blockchain and Federated Learning. *IEEE Internet Things J.* **2021**, *10*, 12112–12125. [CrossRef]
14. Hao, K.; Xin, J.; Wang, Z.; Yao, Z.; Wang, G. Efficient and Secure Data Sharing Scheme on Interoperable Blockchain Database. *IEEE Trans. Big Data* **2023**, *9*, 1171–1185. [CrossRef]
15. Shen, J.; Zhou, T.; He, D.; Zhang, Y.; Sun, X.; Xiang, Y. Block Design-Based Key Agreement for Group Data Sharing in Cloud Computing. *IEEE Trans. Dependable Secur. Comput.* **2017**, *16*, 996–1010. [CrossRef]
16. Zhou, Z.; Tian, Y.; Xiong, J.; Ma, J.; Peng, C. Blockchain-Enabled Secure and Trusted Federated Data Sharing in IIoT. *IEEE Trans. Ind. Inform.* **2022**, *19*, 6669–6681. [CrossRef]
17. Mei, Q.; Yang, M.; Chen, J.; Wang, L.; Xiong, H. Expressive Data Sharing and Self-Controlled Fine-Grained Data Deletion in Cloud-Assisted IoT. *IEEE Trans. Dependable Secur. Comput.* **2022**, *20*, 2625–2640. [CrossRef]
18. Han, D.; Chen, J.; Zhang, L.; Shen, Y.; Wang, X.; Gao, Y. Access control of blockchain based on dual-policy attribute-based encryption. In Proceedings of the 2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Yanuca Island, Cuvu, Fiji, 14–16 December 2020; pp. 1282–1290. [CrossRef]
19. Yan, L.; Ge, L.; Xu, J. Research on data access scheme based on attribute-based encryption in blockchain environment. In Proceedings of the 2023 11th International Conference on Communications and Broadband Networking (ICCBN '23), Xi'an China, 24–26 February 2023; Association for Computing Machinery: New York, NY, USA, 2023; pp. 8–12. [CrossRef]
20. Shen, Y.; Song, W.; Zhao, C.; Peng, Z. Secure Access Control for eHealth Data in Emergency Rescue Case based on Traceable Attribute-Based Encryption. In Proceedings of the 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Wuhan, China, 9–11 December 2022; pp. 201–208. [CrossRef]
21. Rao, Y.S.; Prasad, S.; Bera, S.; Das, A.K.; Susilo, W. Boolean Searchable Attribute-Based Signcryption With Search Results Self-Verifiability Mechanism for Data Storage and Retrieval in Clouds. *IEEE Trans. Serv. Comput.* **2024**, *17*, 1382–1399. [CrossRef]
22. Zhang, K.; Zhang, Y.; Li, Y.; Liu, X.; Lu, L. A Blockchain-Based Anonymous Attribute-Based Searchable Encryption Scheme for Data Sharing. *IEEE Internet Things J.* **2023**, *11*, 1685–1697. [CrossRef]
23. Huang, Q.; Yan, G.; Wei, Q. Attribute-Based Expressive and Ranked Keyword Search Over Encrypted Documents in Cloud Computing. *IEEE Trans. Serv. Comput.* **2023**, *16*, 957–968. [CrossRef]
24. Lewko, A.; Waters, B. Decentralizing attribute-based encryption. In *Advances in Cryptology—EUROCRYPT 2011*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 568–588.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.