electronics

MDPI

Article

A Novel Data Partitioning Method for Active Privacy Protection Applied to Medical Records

Nawaf Alharbe [1,*], Abeer Aljohani [1] and Mohamed Ali Rakrouki [2,3,4]

[1] Applied College, Taibah University, Medina 42353, Saudi Arabia
[2] College of Computer Science and Engineering, Taibah University, Medina 42353, Saudi Arabia
[3] Ecole Supérieure des Sciences Economiques et Commerciales de Tunis, University of Tunis, Tunis 1089, Tunisia
[4] Business Analytics and DEcision Making Lab (BADEM), Tunis Business School, University of Tunis, Ben Arous 2059, Tunisia
* Correspondence: nrharbe@taibahu.edu.sa

Abstract: In recent years, cloud computing has attracted extensive attention from industry and academia due to its convenience and ubiquity. As a new Internet-based IT service model, cloud computing has brought revolutionary changes to traditional computing and storage services. More and more individual users and enterprises are willing to deploy their own data and applications on the cloud platform, but the accompanying security issues have also become an obstacle to the development of cloud computing. Multi-tenancy and virtualization technologies are the main reasons why cloud computing faces many security problems. Through the virtualization of storage resources, multi-tenant data are generally stored as shared physical storage resources. To distinguish the data of different tenants, labels are generally used to distinguish them. However, this simple label cannot resist the attack of a potential malicious tenant, and data still has the risk of leakage. Based on this, this paper proposed a data partitioning method in a multi-tenant scenario to prevent privacy leakage of user data. We demonstrate the use of the proposed approach in protecting patient data in medical records in health informatics. Experiments show that the proposed algorithm can partition the attributes more fine-grained and effectively protect the sensitive information in the data.

Keywords: computer security; data protection; cloud computing; health informatics

Citation: Alharbe, N.; Aljohani, A.; Rakrouki, M.A. A Novel Data Partitioning Method for Active Privacy Protection Applied to Medical Records. *Electronics* **2023**, *12*, 1489. https://doi.org/10.3390/electronics12061489

Academic Editor: KC Santosh

Received: 7 February 2023
Revised: 17 March 2023
Accepted: 20 March 2023
Published: 22 March 2023

1. Introduction

Cloud storage is a crucial byproduct of cloud computing that is gradually penetrating daily life. Regardless of location or time restrictions, cloud storage enables users to access their own data through any networked device at any time. Data storage in cloud storage systems is becoming more and more popular among both individual users and businesses. The amount of data is expanding quickly across a wide range of sectors and application fields as the digital era progresses. However, with the rapid development of cloud computing, its own security problems have become increasingly prominent. The two primary cloud computing technologies, multi-tenancy [1] and virtualization [2], are precisely the cause of the industry's numerous security issues. In the cloud computing environment, "renting" is the most basic business model. Tenants can rent storage resources via virtualization technology. The following is a brief analysis of the main reasons for the security challenges of cloud computing caused by these two key technologies:

1. Virtualization technology virtualizes physical hardware resources and provides them to users in a combined way. This rental method is realized through virtual machines. Multiple virtual machines run on a physical machine and share the same hardware resources. However, through side-channel attacks, traffic analysis, virtual machine escape, and other methods, attackers can still obtain data of other virtual machines from one virtual machine [3].

2. Multi-tenant technology enables multiple tenants in the cloud to use the same application service at the same time, and the storage of multi-tenant data currently mainly adopts the shared database sharing mode [1], this mode of data storage mainly adopts a key-value pair and metadata-driven approach to support multi-tenant database customization and expansion. However, once an attack occurs in this storage mode, it will cause serious data privacy leakage.

These aforementioned security issues affected health informatics, which is one of the quickest-growing sectors in health industries. It entails the skillful application of knowledge and technology to improve patient care. With the advancement of computing technology, especially artificial intelligence and computing hardware, what can be achieved to support patient care via intelligent patient data processing is beyond imagination. However, patient health and care information are confidential, and in most countries, there are very strict regulations around the privacy, confidentiality, and security of such data. With the complexities around the treatment of patients by different healthcare providers, medical, and care practitioners, such data are stored in a distributed manner and/or in the cloud, making patient data privacy protection for intelligent information gathering via data processing a practical significant challenge.

The physical isolation that underlies conventional data storage implements the data isolation of various users (see Figure 1a). The data of many users will not interfere with one another, and each user may have their own storage service. Yet, in a cloud computing environment, many tenants' data are shared and kept in a single storage cluster, and the data isolation between various tenants is logically segregated (see Figure 1b). Hence, if the server is compromised, it will result in a major privacy leakage issue. To not impact the efficiency of application data processing, the data are often stored on the server in plaintext, along with a lot of tenant private data. In addition, regular encryption is no longer appropriate for applications that often access tenant data, such as health informatics. In light of this, the main emphasis of research on multi-tenant data privacy protection is how to effectively safeguard data privacy without compromising the operational effectiveness of systems.

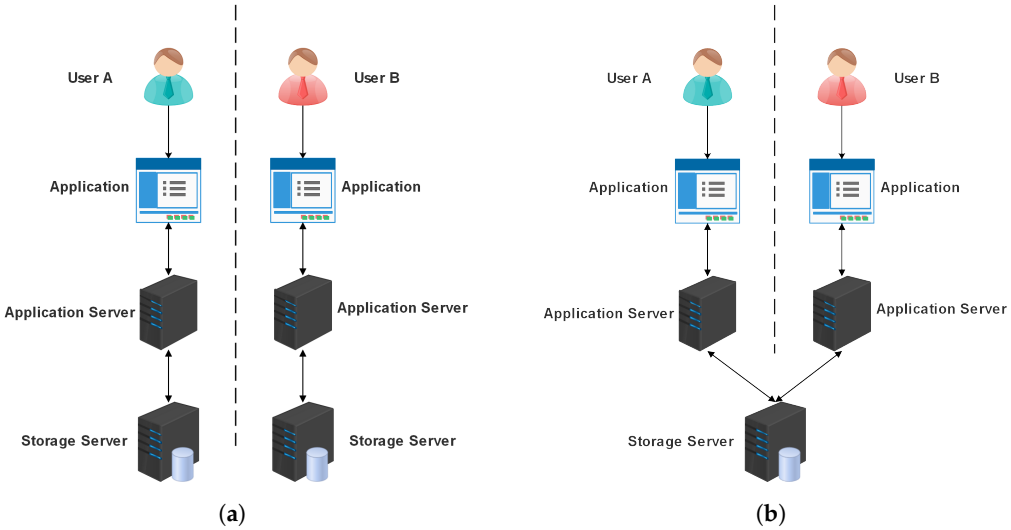


Figure 1. Data storage isolation. (**a**) Physical isolation; (**b**) Logical isolation.

The two major techniques used to protect data privacy are typically encryption and obfuscation. To change the data, encryption usually starts with the goal of concealing the true information included in the data. This alteration is irreversible without a key, making it challenging for an attacker to determine the original data even after obtaining the encrypted data. However, to prevent attackers from brute-forcing the key, traditional encryption methods are designed to be complex, requiring tedious mathematical calculations to com-

plete the data encryption and decryption operations. As an alternative, data obfuscation mostly conceals the original information in the data through anonymity or generalization. Anonymity mainly ensures the equivalence of each data group by adding obfuscated data to the original data to achieve the purpose of hiding information; Generalization mainly hides the original information by expanding the data value, and the extended range is the original data interval, and the original discrete value is expanded into a continuous data interval. Compared with encryption, the calculation speed of data obfuscation is much faster, but there is also a fatal disadvantage of data obfuscation, which is the problem of data reconstruction. When reading data, it takes a lot of energy to remove the dirty data generated during obfuscation. Therefore, obfuscation methods are also not applicable for use with many application scenarios.

To sum up, it is of great significance to both cloud users and cloud providers to study the new challenges of data security storage technology in cloud environments, especially multi-tenancy and virtualization to cloud data security storage. In particular, according to the needs of multi-tenant applications to store data, the research in this paper mainly considers data privacy partition and data processing efficiency tenant privacy data efficiently without encryption.

The rest of this paper is organized as follows. Relevant related works are presented in Section 2. Section 3 introduces the composition of the model, and its optimization and details the algorithmic process of the generative probabilistic model. Section 4 focuses on the experiments, introduces the relevant settings of the experiment, and analyzes the experimental results. Finally, the content of this paper is summarized.

2. Literature Review

In response to privacy protection issues, Karabulut and Nassi [4] proposed a privacy protection method based on the fusion of symmetric encryption technology and asymmetric encryption technology by analyzing the characteristics of SaaS application services and proposed a secure enterprise service based on this method's management structure. Although this method can effectively protect the privacy of user data, the algorithm complexity is too high, and the computational cost is high. Aiming at the problem that tenants need to spend a significant amount of computational cost to decrypt the entire data set in traditional data encryption storage, Fan and Huang [5] proposed a pre-encryption technology by distributing a pre-encryption technology to cloud service providers. The encrypted token generated by the private key downloads the data to be accessed by the tenant in advance before the tenant accesses the data, realizes the search function of ciphertext, and reduces the overhead of decryption. The above two methods must perform encryption and decryption operations before operating on the data. Liu et al. [6] proposed a searchable encryption method. By creating a small index, fuzzy keywords can be retrieved in the state of ciphertext. However, this mechanism is only limited to the retrieval of ciphertext. The operation does not support the retrieval of the data.

Traditional and complex encryption methods consume a significant amount of computing performance and have a great impact on the performance of data access. To solve this problem, researchers proposed a new method to prevent privacy leakage. The most commonly used methods are partition-based methods and k-anonymity-based methods. Sweeney [7] proposed a k-anonymity method, which requires each piece of information to be indistinguishable from other $k-1$ pieces in the divided grouping. To prevent malicious attackers from guessing the connection between private data and identity using background knowledge attacks and consistency attacks, Wong et al. [8] proposed (α, k)-anonymous data partitioning based on k-anonymity method, while ensuring that the k-anonymity principle is satisfied, it requires that the percentage of the amount of information related to private data in each equivalence class divided cannot be greater than a. Machanavajjhala et al. [9] proposed the l-diversity partition method, which requires that the privacy data of each equivalence class have at least l different values so that the attacker can guess the connection between the individual and the sensitive data with a probability

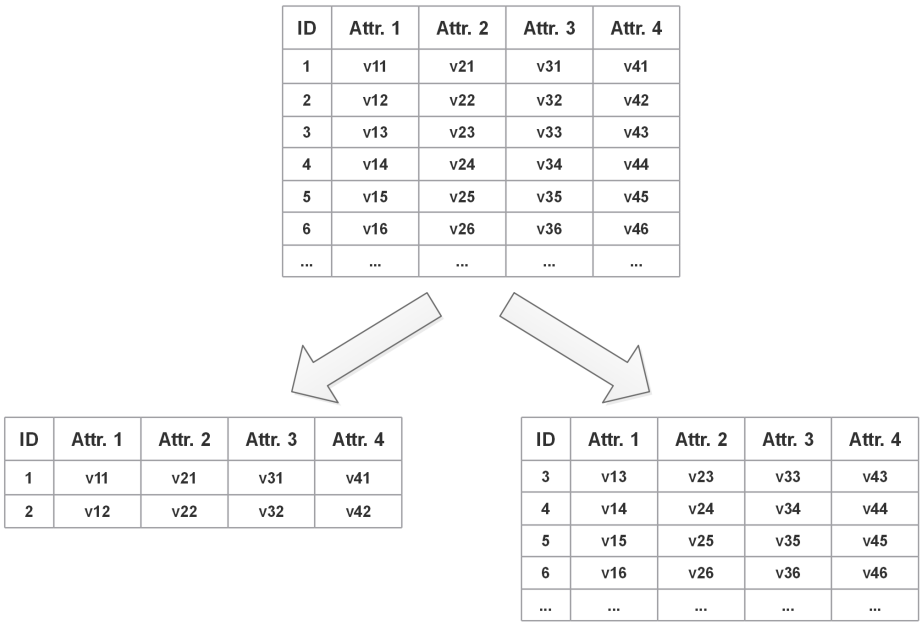
of at most $1/l$. Based on l-diversity, Li et al. [10] proposed the t-closeness principle, which mainly considers the distribution of private data. It requires that the distribution of sensitive data in each equivalence class be as close as possible to the global distribution of the sensitive data. The above-mentioned anonymity-based partitioning methods will cause data distortion. To preserve the original data information to the greatest extent, Wang [11] analyzed the impact of different privacy protection degrees on information loss in detail and proposed a fuzzy data partitioning method. This method defines the ambiguity of privacy and ensures the security of tenants' private data while minimizing the degree of information loss. A comprehensive survey on data security and privacy concerns and data encryption technology in cloud storage systems has been thoroughly reviewed by Yang et al. [12].

Through the above analysis, the academic community has already started a certain study of the privacy protection of multi-tenant data. However, most of the studied scenarios do not consider the frequent reconstruction of the processing data. It is not applicable in the scenario of this work, which considers medical data with high access frequency and multi-tenancy. Indeed, in this paper, the data are application-oriented and business-oriented, and we must take into account both the expense of efficient data access and data reconstruction in addition to privacy protection during data storage. Shao et al. [13] and Shi et al. [14] are similar to the research in this paper. Shao et al. [13] propose the RDFA algorithm, a clustering-based data privacy protection partitioning method. Privacy constraints divide data, but only the number of application connections is considered when dividing, and the cost of connections cannot be guaranteed to be the smallest. The PCPP algorithm proposed by Shi et al. [14] is an attribute association tree method to cluster the associations between attributes and then divide the data according to the privacy constraints. The maximum number of blocks is the goal, but this division method often does not get the optimal solution. The above two works only consider data division and do not consider the problem of data processing efficiency after division.

Through the above analysis, inspired by RDFA and PCPP algorithms, aiming at the problem of information attribute values, this paper proposes a data partitioning method for active privacy protection, which actively protects the attributes with large information entropy, and discusses the table after data partitioning. The efficient access problem and the overall idea of the strategy will be introduced in detail below.

3. The Proposed Approach

In a relational database, data are stored in separate tables. For processing efficiency and load balancing, databases often divide tables into rows. As shown in Figure 2, the traditional database access data row-by-row method cannot protect the privacy of user data well because the data of all user attributes are stored in one row. Once the attacker breaks the database, he can easily obtain information on all attributes of the user. To this end, this paper proposes a vertical division method, as shown in Figure 3. Before data storage, the data are divided vertically according to the active privacy protection strategy proposed in this paper, and the divided data are managed in separate tables.

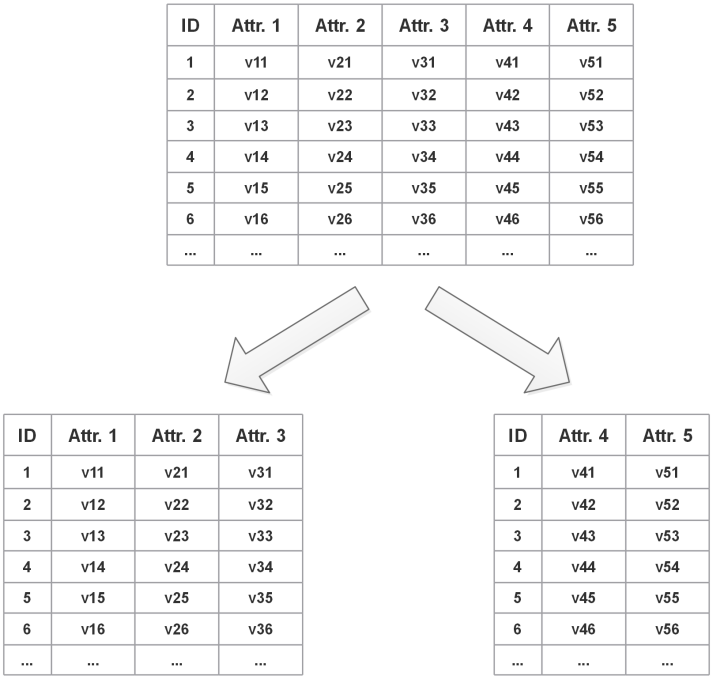


ID	Attr. 1	Attr. 2	Attr. 3	Attr. 4
1	v11	v21	v31	v41
2	v12	v22	v32	v42
3	v13	v23	v33	v43
4	v14	v24	v34	v44
5	v15	v25	v35	v45
6	v16	v26	v36	v46
...	...	...	...	...

ID	Attr. 1	Attr. 2	Attr. 3	Attr. 4
1	v11	v21	v31	v41
2	v12	v22	v32	v42

ID	Attr. 1	Attr. 2	Attr. 3	Attr. 4
3	v13	v23	v33	v43
4	v14	v24	v34	v44
5	v15	v25	v35	v45
6	v16	v26	v36	v46
...	...	...	...	...

Figure 2. Row partitioning.



ID	Attr. 1	Attr. 2	Attr. 3	Attr. 4	Attr. 5
1	v11	v21	v31	v41	v51
2	v12	v22	v32	v42	v52
3	v13	v23	v33	v43	v53
4	v14	v24	v34	v44	v54
5	v15	v25	v35	v45	v55
6	v16	v26	v36	v46	v56
...	...	...	...	...	...

ID	Attr. 1	Attr. 2	Attr. 3
1	v11	v21	v31
2	v12	v22	v32
3	v13	v23	v33
4	v14	v24	v34
5	v15	v25	v35
6	v16	v26	v36
...	...	...	...

ID	Attr. 4	Attr. 5
1	v41	v51
2	v42	v52
3	v43	v53
4	v44	v54
5	v45	v55
6	v46	v56
...	...	...

Figure 3. Column partitioning.

The active privacy protection strategy proposed in this paper mainly includes two parts; the first part is data preprocessing. To minimize the impact of data partitioning on application processing efficiency, the database transactions are first mined for frequent itemsets to find attribute subsets with large correlations. The second part is an active privacy protection strategy. By introducing attribute information entropy, privacy is actively protected, and data are divided according to privacy constraints and active protection policies. The overall scheme is shown in Figure 4.

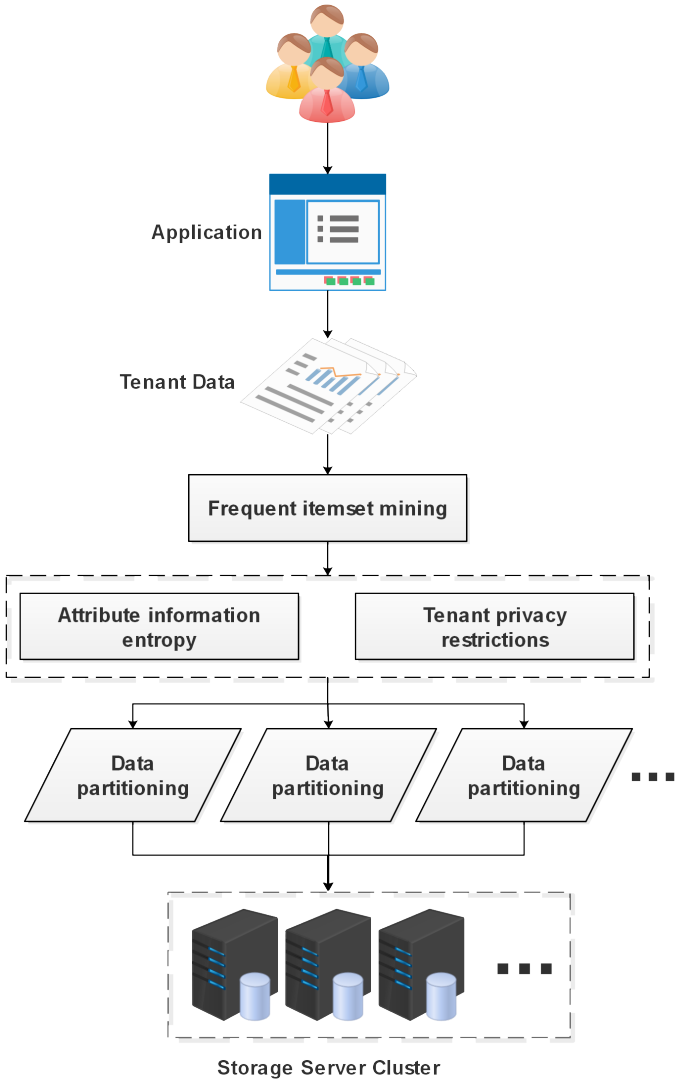


Figure 4. Data partitioning method for active privacy protection.

Through the above description, the data partitioning method of active privacy protection proposed in this paper can effectively protect user data privacy compared with traditional privacy protection methods and can also ensure the efficiency of application access.

3.1. Data Partitioning

Between security and efficiency, we try to find a balance. The goals of data partitioning for active privacy protection are:

1. Consider the SLA (Service-Level Agreement) requirements of different tenants, and divide the data without losing the original data information to achieve the purpose of privacy protection;
2. It can quickly locate and reconstruct the original data, and the data division has the least impact on the application access data.
3. The attributes of the divided subsets are disjoint.

To achieve the above goals, this paper mines the frequent itemsets of database operation transaction rows by constructing a vertical data segmentation strategy and obtains closely related attribute sets. For these attribute sets, they are divided according to the active protection strategy to process different information data values and proactively protect data with great information value.

3.2. Related Definitions

Definition 1. *Privacy Constraint NCC$\{a_1, a_2, \ldots, a_i\}$: Tenant's SLA requirements, indicating that attributes a_i and a_j cannot be stored together.*

Here privacy constraints mainly include three aspects [1]: composition privacy constraints, value privacy constraints, and dependency privacy constraints. Combination privacy constraints, that is, privacy combinations that cannot appear in a data block at the same time, such as name and ID number cannot appear at the same time; value privacy constraints mainly refer to certain values in an attribute that belongs to privacy, such as disease attributes, for a certain attribute. Some diseases, such as colds, do not have privacy, while sensitive diseases, such as AIDS, have privacy and need to be protected separately; relying on privacy attributes mainly means that individual attributes do not have any sensitive information, but together they will reveal the tenant's information. Privacy, such as the doctor's main disease and the patient's name together, can infer the patient's disease.

Definition 2. *Privacy attribute partitioning: The attribute set of the data are divided into different subsets, and the attributes between each subset are not repeated, and the division itself does not violate the tenant's privacy constraints.*

Definition 3. *Attribute association: Indicates the frequency of access between different attributes $(a_1, a_2, a_3, \ldots, a_n)$.*

For example, given a table B in a database, its historical operation record is $D = \{op_1, op_2, op_3, \ldots, op_n\}$, which consists of several operations, such as query, add, etc., a_i represents an attribute in table D, also known as the item, then the attribute association degree $frequency(a_1, a_2, a_3, \ldots, a_n)$ is the operand of the attribute $(a_1, a_2, a_3, \ldots, a_n)$.

As shown in Table 1, the first row indicates that the attribute set involved in the operation op_1 is ABC, and the number of occurrences is 10. The remaining rows are similar. Then, $frequency(A, B, C) = 10$, and $frequency(A, B) = 60$.

Table 1. Operation Frequency Statistics.

Operation	Attribute Collections in Operations	Frequency
op_1	ABC	10
op_2	ABCD	20
op_3	CD	40
op_4	ABE	30

3.3. Active Protection Strategy

To prevent malicious attackers from guessing the connection between sensitive data and user identity through background knowledge attacks, this paper proposes an active protection strategy to protect the privacy of data through the information value of data. According to the information entropy theory proposed by Shannon [15], we can know that if a transaction or information has a larger possible value, it means that it contains more information, and then it is more likely to identify the data record uniquely. In extreme cases, such as ID card numbers, the information entropy is very large and can uniquely identify a person. In addition, the larger the possible value range of an attribute, the more useful information may be obtained if a malicious attacker obtains the data. For example, an attribute of the data is "marital status"; if its value range is "married" or "unmarried", then it can be used as non-sensitive data. If its value range is "married", "divorced", "single", "widowed", "separated", etc., this attribute is very likely a sensitive attribute. For example, a certain attribute of hotel data is the user's room opening time. Different users have different room opening times, its value range is wide, and the information entropy is large. If an adversary obtains this data, although it is impossible to know which records

correspond to it, the adversary can calculate the operation of the hotel, user behavior preferences, and other information based on this time. It can be seen that the value range of an attribute is one of the keys to determining whether an attribute is a privacy attribute. Based on this, this paper proposes the concept of attribute information entropy. Through value range analysis, an active protection strategy is implemented for attributes whose information entropy exceeds a certain threshold. Attributes with large information entropy cannot be divided together.

Definition 4. *Attribute information entropy $H(x)$: Represents the informational value of a data attribute. Its value is determined according to the value range of the attribute. The larger the value range, the greater the amount of information it contains and the greater the information entropy, where p_i represents the probability that the attribute takes the value i, and n represents the value range of the attribute.*

$$H(x) = -\sum_{i=1}^{n} p_i \ln p_i \tag{1}$$

Definition 5. *Attribute information entropy threshold $H(x)_{th}$: When the information entropy of an attribute in the table is greater than the threshold, that is, $H(x) > H(x)_{th}$, this paper considers that the attribute has privacy attributes.*

Definition 6. *Penalty factor σ: In the excavated frequent itemsets, we "penalize" the attributes whose attribute information entropy $H(x) > H(x)_{th}$ because such attributes are not allowed to be grouped together.*

$$Penality = \sigma H(x) \tag{2}$$

According to the above definition, it can be concluded that the privacy constraints in this paper mainly come from two aspects. On the one hand, it represents the privacy protection requirements of the tenant SLA. For example, if the data of a tenant has two attributes, name, disease, then the tenant does not want their disease privacy to be leaked, so the two attribute data should be placed separately; on the one hand, it is determined by the information entropy of the attribute. When the information entropy is greater than the threshold, it also becomes a privacy attribute. Specific steps are as follows:

1. Calculate the information entropy of each attribute according to Equation (1);
2. Determine whether the information entropy of each attribute is greater than the threshold;
3. If it is greater than the value, the attribute will be "penalized": in the frequent itemset excavated, the value corresponding to the operation containing the attribute is subtracted from the value of punch according to Equation (2), and the frequent item containing the attribute is set. Let the value of the frequent item containing this attribute be a_i, then the value a_i after the penalty is:

$$a_i = a_i - Penality \tag{3}$$

4. If it is less than, do nothing;
5. Update the frequent itemset table.

3.4. Data Partitioning Method

To minimize the impact of the divided data on the application access efficiency, in the process of data division, data with large attribute associations should be divided together as much as possible and, at the same time, meet the requirements of privacy constraints. To achieve this goal, the algorithm uses attribute association degree to mine frequent itemsets of attributes. After dividing by privacy constraint rules, the remaining attributes with the largest association degree are divided into a data block. The algorithm is mainly divided into 3 steps:

1. Frequent itemsets mining based on operation frequency: The algorithm uses the FP_Growth mining algorithm [16] to mine the frequent items of the data and obtains the frequent itemsets of all attribute operations. FP_Growth algorithm is an association rule algorithm based on frequent itemsets proposed by [17]. It can get the frequent itemsets of the data through 2 scans, which is efficient, so this paper uses FP_Growth to mine the frequent itemsets of the data.
2. Frequent item value "penalty" based on attribute entropy: First, calculate the information entropy $H(x)_i$ of each attribute, and judge whether the value of its attribute entropy is greater than the threshold. If it is greater than Equation (2), then the value of the attribute contained in the frequent itemset is processed, and the newly obtained frequent itemsets are used to partition the data.
3. Data partitioning based on privacy constraints: According to the tenant's privacy constraints, delete the frequent items containing the constraints from the set, select the largest frequent items from the remaining frequent items and store them together, and judge that the remaining attributes can be merged. If so, merge them into a set. If not, store them separately.

The specific algorithm flow is shown in Algorithm 1.

Algorithm 1 Data partitioning algorithm

Initialize: Attribute Operation Frequency Set AOFS (Attribute Operation Frequency Set)
Output: Attribute Partition Set APS (Attribute Partition Set)

1: Use the FP_Growth mining algorithm to mine the frequent items of the data and obtain the frequent item sets FIS (Frequent item sets) of all attribute operations;
2: Calculate the information entropy of all attributes by Equation (1), compare the information entropy with the threshold, and put the attributes whose information entropy is greater than the threshold into the penalty queue L;
3: Determine whether L is empty; if it is empty, jump to (5)
4: Take the team head attribute i of L, search for the frequent item containing attribute i in the FIS, process the value of the frequent item with Equation (2), and update the FIS; jump to (3);
5: According to the privacy constraints of the tenants, delete frequent items containing privacy constraints in the FIS, and update the FIS;
6: Determine whether the FIS is empty; if it is empty, jump to (9);
7: Select the attributes contained in the largest frequent item set in the FIS, divide it into a data division, and delete the frequent items containing these attributes from the FIS;
8: Determine whether the remaining frequent itemsets can be merged. If so, merge them into one data division. If not, divide them and jump to (6);
9: End.

The following is an example to illustrate the data partitioning process proposed in this paper. Assuming that the attribute set of a user data is $\{A, B, C, D, E\}$, the frequent itemsets mined by the FP_Growth mining algorithm are shown in Table 2. The tenant's privacy constraint is $\{A, B\}$; that is, attribute A and attribute B cannot be stored together.

Table 2. Frequent itemsets.

Candidate Itemset	Frequency of Operation
$\{A, B, C\}$	25
$\{A, C, E\}$	41
$\{A, B, E\}$	25
$\{B, C, D\}$	40

According to the tenant's privacy constraints, delete the frequent items that include attribute A and attribute B in the frequent itemset, and obtain the frequent itemset as shown in Table 3.

Table 3. Frequent itemsets filtered by privacy constraints.

Candidate Itemset	Frequency of Operation
$\{A, C, E\}$	41
$\{B, C, D\}$	40

Assuming that the calculated attribute information entropy is 0.5, 0.6, 1.6, 0.7, 1.2, the penalty factor σ is set to 10, and the information entropy threshold is set to 1, then it can be obtained that the information entropy of attribute C and attribute E exceeds the threshold, then the system determines that it is a privacy attribute and needs to be protected. At this time, if the active protection strategy is not implemented, then for the results in Table 4, attributes $\{A, C, E\}$ should be divided into a data division, but we found that the information entropy of attribute C is 1.6, and the information entropy of E is 1.6. The information entropy is 1.2, and the information they contain is of great value. If attributes C and E are stored together, attackers can easily guess valuable information through these two attributes, so this division method is unreasonable.

"Penalize" the frequent items that contain attribute C and attribute E in the frequent itemset, and the frequent itemset obtained becomes as shown in Table 4.

Table 4. Penalized frequent itemsets.

Candidate Itemset	Frequency of Operation
$\{A, C, E\}$	13
$\{B, C, D\}$	24

According to the result, attributes $\{B, C, D\}$ should be divided into a data partition for storage, and the rest are attribute A and attribute E. Since there is no attribute constraint between them, attributes A and E are combined and stored into a data partition. Such a division method avoids dividing attributes with high attribute information entropy into one table and actively protects attribute privacy.

3.5. Confusion of Data Partition Correspondence

Data integrity and data confidentiality are basic requirements for partition-based privacy protection. Data confidentiality means that the data block satisfies all privacy constraints, and the method proposed in this paper can satisfy the data confidentiality requirement. Data integrity requires that the original data blocks be able to reconstruct the original data relational table, and the reconstruction of the data are carried out through the ID in the table. From Figure 3, we can see that if only the original ID is used for processing, it is easy to obtain the corresponding relationship between the records of each table, which cannot achieve the purpose of privacy protection. Therefore, it is necessary to confuse the correspondence between data divisions.

Definition 7. *Data partition mark, which means the unique mark that the same record is divided into different tables, which is represented by TID (Table ID).*

In the vertical division of data, the same record is divided into several different parts and stored in different tables for management. When reading data, it is necessary to reconstruct the data into a complete record, which requires recording these Correspondences between different records in the table. In addition, the same records should be marked differently in different tables; otherwise, the correspondence between data records will also be leaked. Different tenants should have different obfuscation results and, at the same time, reflect the characteristics of tenant customization; the obfuscated tags cannot affect

the operation of the database. To sum up, this paper uses a hash function [18] to generate tags recorded in different tables, as shown in Equation (4).

$$TID = hash(ID \cdot t_i) \tag{4}$$

where *hash* is the hash function selected in the tenant privacy protection policy, ID is the ID number of the record in the original table, and t_i is the identifier of the table to be stored. After the table is divided through the above steps, the TID and the original ID are put into the HashMap so that the original data can be quickly queried and reconstructed. There are various methods for the protection of HashMap, which can be encrypted and stored in the server or stored in a trusted third party.

The above method can make the marks of the same record in each table different, achieve the purpose of confusing the corresponding relationship of data division, and protect the privacy relationship between data tables. Even if an attacker tries to brute force the corresponding relationship between different data tables, it is impossible to determine whether the corresponding relationship is correct. The data partitioning method of active privacy protection proposed in the paper can ensure privacy protection in the same data table, and attackers cannot analyze and guess the privacy of tenants even if they get any of the tables. Through this double guarantee of "internal" and "external" relationships, the proposed method can effectively protect the privacy of tenant data.

4. Experimental Results

4.1. Experimental Environment

For the division of private data, we mainly test the rationality of the partitioning strategy, the division efficiency and the number of application connections. The proposed algorithm in this paper (named Active_Protect) is compared with the PCPP algorithm [13], RDFA algorithm [14], and a random division method for the rationality of division, and the division efficiency and connection times

The experiment is implemented with Java code, the database uses MySQL, and the code is written under a Windows 10 platform. The hardware configuration is an Intel i7-based PC with 16 GB of RAM.

4.2. Data Sources

The data used in this experiment comes from the medical insurance data of a project in the laboratory and the corresponding system log. The test data were randomly sampled from a larger database of about 50,000 records. The data is the single disease data of a certain disease. For the operation records of the database, we randomly select 50,000 records from the system log as the experimental test data. The attributes of the data are represented by $Attrs\{a_1, a_2, a_3, \ldots, a_n\}$, where the privacy constraint $NCC\{a_i, a_j\}$ means that the attribute a_i cannot be placed with the attribute a_j. The randomly sampled experimental data are described in detail below.

Due to a large number of data attributes, this paper only selects a part of the representative attribute data for testing. The specific attributes and attribute descriptions of the data are shown in Table 5. Because it involves a lot of personal privacy, part of the processing that contains obvious personal information is done. In fact, the experimental data are also processed during the test process.

Input: medical insurance record data for a certain disease.

Table 5. Medical insurance record data for a certain disease.

Attribute #	Attribute Name	Attribute Description
A1	Name	Name of the patient
A2	Gender	Gender of the patient
A3	Date of Birth	Date of Birth of the patient
A4	ID	The number of this visit
A5	The minimum treatment time	The time of the first treatment of the patient
A6	Treatment Years	The patient treatment years in the data record
A7	the number of treatments in the first quarter	
A8	The number of treatments in the second quarter	
A9	The number of treatments in the third quarter	
A10	The number of treatments in the fourth quarter	
A11	Treatment hospital number	The ID of the hospital
A12	Hospital Name	The name of the treatment hospital
A13	Total medical expenses	The total expenses spent this time
A14	Drug fee	The total cost of this drug
A15	Number of visits	Number clinic visits
A16	Fund reimbursement amount	Basic reimbursement amount
A17	Amount of subsidy for serious illness	Amount of subsidy for serious illness

For the sake of simplicity, this paper sets the user's privacy protection constraints on the disease as shown in Table 6.

Table 6. User Privacy Restrictions.

Privacy Constraints
NCC_1 {A1, A5}
NCC_2 {A1, A6}
NCC_3 {A1, A16}
NCC_4 {A1, A17}
NCC_5 {A1, A2, A3}

4.3. Experimental Results Analysis

According to the privacy protection partitioning method proposed in this paper, the information entropy of each attribute is first calculated. The calculation results and related calculation methods are shown in Table 7.

The system enables log statistics to record the fields involved in each database operation. For the operation log, this paper preprocesses it. We do not care about the value of the data and the type of operation, we care about the properties involved in the operation.

According to the statistical results of a period of time, the frequent itemsets are mined for the operation times of attributes. In this paper, the threshold of frequent items is set to 40,000 times, and the penalty factor σ is 5000. At the same time, this paper also compares the RDFA and PCPP algorithms. For the same input data, the division results obtained are shown in Table 8. From the table, we can see that the RDFA algorithm divides the attribute into 4 parts, the PCPP algorithm divides the attribute into 3 parts, and the Active_Protect algorithm divides the attribute into 6 parts, and the division of each algorithm can meet the privacy constraints. In terms of division granularity, the algorithm proposed in this paper has the smallest division granularity because this algorithm considers the information entropy of attributes and actively protects those with higher information entropy, so the division has more constraints and the finer the division granularity. From Table 7, it can be seen that the attributes with higher entropy include name, ID, date of birth, minimum treatment time, total medical expenses, fund reimbursement amount, and serious illness subsidy amount. Most of the algorithms proposed in this paper are divided into different divisions, while both RDFA nor PCPP algorithms do not take this into account. For example, in the PCPP algorithm, the total medical expenses, the fund reimbursement amount, and the critical illness subsidy amount are divided together, then the attacker can use these data to calculate the abnormal situation in the medical insurance reimbursement, so the division method proposed in this paper effectively protects the privacy of user data.

Table 7. Attribute information entropy calculation results.

Attributes	Entropy	Calculation Method Description
A1	4.7	The entropy is calculated incrementally. Among the 50,000 pieces of data, except those with the same name, the rest is the value range of name.
A2	0.3	There are only two values for gender, so the value range is 2
A3	4.7	The entropy is calculated incrementally. In the 50,000 pieces of data, except those born in the same year and month, each value has an equal probability of occurrence.
A4	4.7	Each user has a different ID number, so for 50,000 pieces of data, the value range of the ID is 50,000
A5	4.7	Calculated in the same way as the date of birth
A6	1.7	Calculated in the same way as the date of birth
A7	1.7	Calculate the frequency of occurrences of each quarter in the 50,000 pieces of data
A8	1.7	The number of treatments in the same quarter
A9	1.7	The number of treatments in the same quarter
A10	1.7	The number of treatments in the same quarter
A11	1.3	Each hospital number is unique and has an equal probability of occurrence
A12	1.3	Calculated incrementally
A13	3.3	For the calculation of the cost, the unit of this article is selected to one hundred, and the data below ten digits are ignored.
A14	2.7	The calculation method of the medicine fee is the same as the total medical expenses
A15	0.6	
A16	3.0	the same as total medical expenses
A17	2.4	the same as total medical expenses

Table 8. Attribute partitioning results.

Algorithm	Privacy Division Results
Active_Protect	{A1} {A2, A4} {A13, A5} {A15, A16} {A3, A17, A11} {A6, A7, A8, A9, A10, A14, A12}
RDFA	{A1, A4}; {A2, A15, A3}; {A6, A14, A17, A16, A13}; {A5, A7, A8, A9, A10, A11, A12}.
PCPP	{A1, A4, A15}; {A2, A3, A13, A16, A11, A12, A17}; {A6, A5, A7, A8, A9, A10, A14};

4.4. Analysis of Number of Connections

In terms of the number of connections, the above-mentioned operation logs are used to analyze the average number of connections for each algorithm. If the attributes of a record design are in n partition sets, it is recorded as requiring n links. The number of links of the ith record is recorded as n_i, then the average number of links P can be recorded as shown in the equation:

$$P = \sum_{i=1}^{k} \frac{n_i}{k} \tag{5}$$

The average number of links of the three algorithms is compared. As can be seen in Figure 5, the number of links of the RDFA algorithm is the least because it considers the frequency of 2-itemsets when clustering. The number of links of the PCPP algorithm is second because it is the greedy algorithm used in the division, the higher the attribute

association, the bigger, the better, the random division algorithm has the highest evaluation connection times, and the algorithm in this paper takes the attribute information entropy into consideration and actively protects the attributes with high information entropy. The division granularity is finer, so the number of connections is higher than that of the RDFA algorithm but lower than that of the PCPP algorithm, which is within an acceptable range. Because the Random algorithm is randomly divided and does not consider the association between attributes, the average number of connections is the highest.

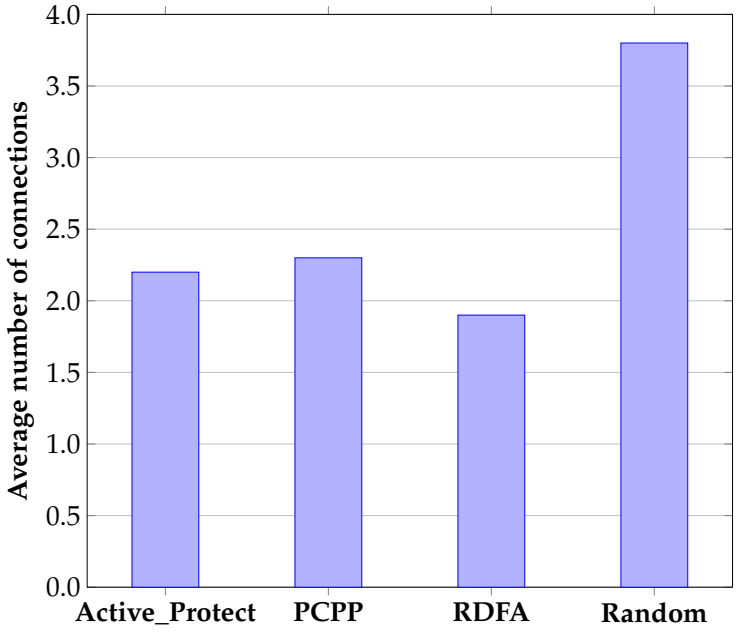


Figure 5. Average number of connections to the application.

4.5. Analysis of Partition Efficiency

In terms of division efficiency, the experiment sets 7 groups of different data attributes. The experiment compares the division efficiency of the Active_Protect algorithm, RDFA algorithm, PCPP algorithm, and random division algorithm under a different number of privacy constraints. The experimental results are shown in Figure 6.

As can be seen from Figure 6, as the number of attributes increases, the time for privacy partitioning increases accordingly, and the efficiency gradually decreases. The random partitioning algorithm consumes the least time because it only considers the tenant's privacy constraints, while other algorithms consume the least time. Because the association between attributes is considered and processed, the privacy partition time is higher than the random partition algorithm. Among the Active_Protect algorithm, RDFA algorithm and PCPP algorithm, when the number of attributes is small, the Active_Protect algorithm has the least division time because the frequent itemset mining algorithm has high efficiency in processing a small amount of data, but with the increase in the number of attributes Increase, the more associations between attributes, the more complex the relationship, the processing efficiency of the frequent itemset mining algorithm is reduced, and the RDFA algorithm uses clustering to find the association between attributes, so it has better performance when the number of attributes increases, the RDFA algorithm consumes a lot of time because of its complex attribute association tree construction, and with the increase in the number of attributes, its consumption time increases almost exponentially.

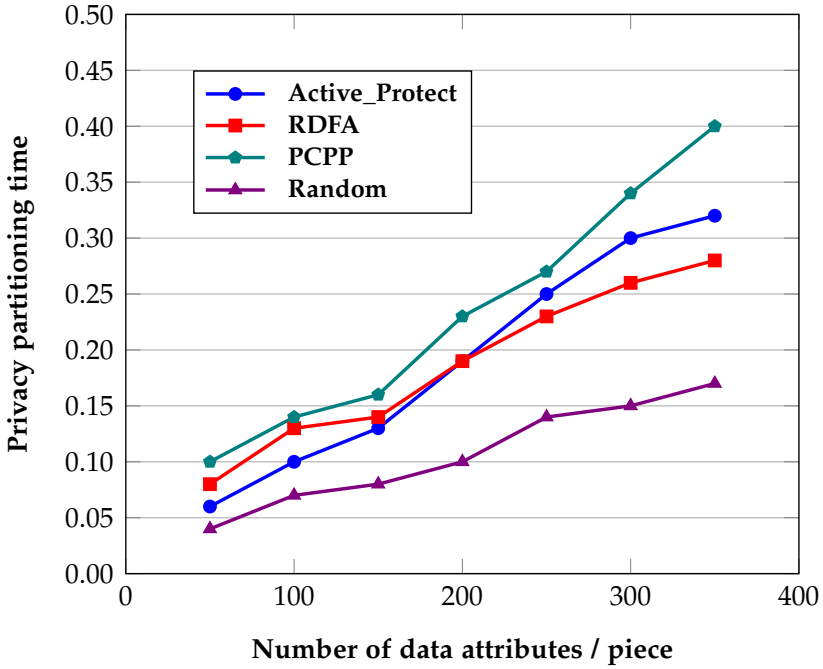


Figure 6. Comparison of privacy partitioning time as the number of privacy constraints increases.

To sum up, the Active_Protect algorithm proposed in this paper has high privacy partition efficiency when the number of attributes is small, and the privacy partition efficiency decreases when the number of attributes is large. In the actual scene, the number of attributes of the data reaches 300, which is already a very high dimension, so the algorithm proposed in this paper has a higher practical application value.

Through experimental comparative analysis, we can see that the algorithm proposed in this paper has more fine-grained attribute division than the RDFA algorithm and PCPP algorithm, which can better protect the privacy of data, and the privacy division time has a good performance when the number of attributes is low. Although the performance on the number of connections is slightly lower than that of the RDFA algorithm, it is still within an acceptable range. Therefore, the active protection of privacy data partitioning method proposed in this paper has certain feasibility.

5. Conclusions

This paper first analyzes the security problems faced by multi-tenant data in shared storage and the limitations of existing privacy protection methods in multi-tenant scenarios. Subsequently, it analyzes the management methods of database tables in detail and proposes an active privacy protection approach for data. The proposed data division-based method uses the information entropy of the attributes to actively protect the attributes with high entropy so as to avoid dividing data with high entropy together and, thus, protect the data privacy while ensuring the efficiency of the system. This paper also considers the confusion of the corresponding relationship of data division and prevents privacy leakage by confusing the corresponding relationship of records between tables. Finally, the experiment based on patient health records is used to prove the feasibility of the algorithm. The proposed active privacy protection approach is ideal to be used within health informatics.

The suggested approach in this paper also has some drawbacks that should be addressed and refined further. The following aspects will be the primary areas of future research:

- After the attribute entropy is penalized, it is possible that the two biggest attribute entropies are still combined into one data block in the privacy-preserving data division technique; however, this topic is not addressed in this study, and the information entropy calculation is also not included. The computation of information entropy can

be made faster since in each attribute's value range needs to be counted, and the time required is too great.

- Traditional access control technologies use static policy rules to limit user requests, but as the dynamic business environment of the cloud changes, especially in terms of the security and integrity of the environment for resource access, it poses a security risk to the access control system that a static access control cannot properly address. A risk-based access control model that is data-oriented might be the subject of future study. Such studies could strengthen the approach proposed in this work regarding securing data access.

Author Contributions: Conceptualization, N.A. and M.A.R.; methodology, N.A. and A.A.; validation, N.A. and M.A.R.; formal analysis, N.A., M.A.R. and A.A.; investigation, A.A. and M.A.R.; resources, N.A. and A.A.; data curation, N.A., A.A. and M.A.R.; writing—original draft preparation, N.A. and M.A.R.; writing—review and editing, N.A. and A.A.; visualization, N.A., A.A. and M.A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used and analyzed during the current study are available from the corresponding author upon reasonable request.

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

References

1. Shen, Y.; Cui, W.; Li, Q.; Shi, Y. Hybrid fragmentation to preserve data privacy for SaaS. In Proceedings of the 8th Web Information Systems and Applications Conference, Chongqing, China, 21–23 October 2011; pp. 3–6. [CrossRef]
2. Prasath, M.A.; Sathiyabama, T. Virtualization in Cloud Computing. *Int. J. Trend Sci. Res. Dev.* **2018**, *2*, 875–877. [CrossRef]
3. Alouffi, B.; Hasnain, M.; Alharbi, A.; Alosaimi, W.; Alyami, H.; Ayaz, M. A Systematic Literature Review on Cloud Computing Security: Threats and Mitigation Strategies. *IEEE Access* **2021**, *9*, 57792–57807. [CrossRef]
4. Karabulut, Y.; Nassi, I. Secure enterprise services consumption for SaaS technology platforms. In Proceedings of the International Conference on Data Engineering, Shanghai, China, 29 March–2 April 2009; pp. 1749–1756. [CrossRef]
5. Fan, C.I.; Huang, S.Y. Controllable privacy preserving search based on symmetric predicate encryption in cloud storage. *Future Gener. Comput. Syst.* **2013**, *29*, 1716–1724. [CrossRef]
6. Liu, C.; Zhu, L.; Li, L.; Tan, Y. Fuzzy keyword search on encrypted cloud storage data with small index. In Proceedings of the 2011 IEEE International Conference on Cloud Computing and Intelligence Systems, Beijing, China, 15–17 September 2011; pp. 269–273. [CrossRef]
7. Sweeney, L. k-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* **2002**, *10*, 557–570. [CrossRef]
8. Wong, R.C.W.; Li, J.; Fu, A.W.C.; Wang, K. (α, k)-anonymity: An enhanced k-anonymity model for privacy-preserving data publishing. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, 20–23 August 2006; Volume 2006, pp. 754–759.
9. Machanavajjhala, A.; Gehrke, J.; Kifer, D.; Venkitasubramaniam, M. *l*-diversity: Privacy beyond *k*-anonymity. *ACM Trans. Knowl. Discov. Data* **2007**, *1*, 24–24. [CrossRef]
10. Li, N.; Li, T.; Venkatasubramanian, S. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering, Istanbul, Turkey, 15–20 April 2007; pp. 106–115. [CrossRef]
11. Wang, H. Privacy-preserving data sharing in cloud computing. *J. Comput. Sci. Technol.* **2010**, *25*, 401–414. [CrossRef]
12. Yang, P.; Xiong, N.; Ren, J. Data Security and Privacy Protection for Cloud Storage: A Survey. *IEEE Access* **2020**, *8*, 131723–131740. [CrossRef]
13. Shao, Y.; Shi, Y.; Li, H. A Novel Cloud Data Fragmentation Cluster-based Privacy Preserving Mechanismk. *Int. J. Grid Distrib. Comput.* **2014**, *7*, 21–32. [CrossRef]
14. Shi, Y.; Jiang, Z.; Zhang, K. Policy-Based Customized Privacy Preserving Mechanism for SaaS Applications. In *Grid and Pervasive Computing*; Park, J.J.J.H., Arabnia, H.R., Kim, C., Shi, W., Gil, J.M., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 491–500.
15. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]
16. Zhou, L.; Zhong, Z.; Chang, J.; Li, J.; Zhexue, J.; Feng, S. Balanced parallel FP-growth with mapreduce. In Proceedings of the 2010 IEEE Youth Conference on Information, Computing and Telecommunications (YC-ICT 2010), Beijing, China, 28–30 November 2010; pp. 243–246. [CrossRef]

17. Pei, J.; Han, J.; Mortazavi-Asl, B.; Wang, J.; Pinto, H.; Chen, Q.; Dayal, U.; Hsu, M.C. Mining sequential patterns by pattern-growth: The prefixspan approach. *IEEE Trans. Knowl. Data Eng.* **2004**, *16*, 1424–1440. [CrossRef]
18. Zhang, K.; Li, Q.Z.; Shi, Y.L. Research on Data Combination Privacy Preservation Mechanism for SaaS. *Chin. J. Comput.* **2010**, *33*, 2044–2054. [CrossRef]