Zhao et al. EURASIP Journal on Wireless Communications and Networking (2018) 2018:281
https://doi.org/10.1186/s13638-018-1290-3

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Research on symmetric fuzzy search of medical data outsourcing system under intelligent network

Huiqi Zhao[1,2], Qian Chen[1], Yinglong Wang[2*] and Minglei Shu[2]

Abstract

Intelligent networks have developed rapidly in various fields of the world today. As a popular network technology, they have broad application prospects in the fields of telemedicine and health monitoring. The application of this technology has important practical significance for solving many aspects of medical treatment. Aiming at the encryption scheme of fuzzy keyword search and the verifiable problem of the data returned by medical cloud in the security of medical cloud data, this paper analyzes the encryption of medical data outsourcing system on medical cloud and the security search method. A verifiable fuzzy keyword symmetric searchable encryption scheme for medical cloud data outsourcing system is proposed, which not only uses symmetric encryption algorithm, but also supports the search of fuzzy keyword and supports the verification of search results. So that the confidentiality of medical data is guaranteed, medical data administrators in the medical outsourcing data uploaded to the medical cloud server before the medical outsourcing data encryption, when patients or other medical users need to obtain a file, through keyword query to the medical cloud to make a request, so as to ensure that their privacy. It also ensures the integrity and security of the returned results.

Keywords: Intelligent network, Medical cloud server, Symmetrical model, Verifiability, Medical outsourcing system, Searchable encryption, Index, Security, Trapped door

1 Introduction

With the development of network applications in the direction of depth, the network has become the lifeline of the enterprise information system, and the information intelligent network is to optimize the existing network infrastructure, so that the new technology applications such as wireless, storage, and intelligent transmission are added to the existing network conveniently and effectively with the needs of the business. Taking medical remote medical monitoring as an example, wireless wearable medical monitoring has become possible. According to the requirement, people can test and transmit data through various sensors on the body, need to form a wireless network structure, and use smart sensors and send the data to get into the human body health, sports, and other conditions; to obtain what we want in medical data, through the wireless body in patients with physical layout domain network, the medical workers need various physiological parameters transmitted by wireless monitoring instrument, so that we can avoid the influence of the instrument line, solve the instrument's influence on the patients' activity space, and also lighten the load of the medical staff 24-h monitoring patients and work data records can be complete and correct. In addition, the long-term monitoring status of wireless monitoring system plays a role in preventing disease during the accumulation of pathological data. For normal healthy people, health care can also be carried out through such a portable monitoring system. At the same time, in some specific groups of people, such as athletes, by monitoring the rhythm of the heart and the information such as temperature and speed strength, to prompt the control training intensity, athletes in the invisible monitoring system also have played an important role in health and fitness coach. With the maturity and development of

* Correspondence: wangylscsc@126.com
[2]Shandong Provincial Key Laboratory of Computer Networks, Qilu University of Technology (Shandong Academy of Sciences), Shandong Computer Science Center (National Supercomputer Center in Jinan), Shandong Jinan 2750014, China
Full list of author information is available at the end of the article

technology, intelligent network will play an important role in life, medical, entertainment, military, and other fields and applications. According to our research, intelligent networks can be roughly divided into two categories: medical applications and non-medical applications. At present, more and more scholars and experts in the field of intelligent network research, with the need for more and more applications and limitations, the idea of intelligent network system, and architecture ideas put forward new requirements and challenges. And with the rapid popularization of cloud computing, more and more individuals and enterprise users choose to send their information to the cloud, for example, in medical, medical drug data, patient treatment records, and other sensitive data files, the storage of medical cloud can not only reduce the cost of medical data maintenance and backup, but can also provide high-quality data on-demand storage services for medical data administrators (medical data administrators), thereby reducing the burden of data administrators on the storage and maintenance of patient data. However, because the medical staff or patients and the medical cloud server is not in the same trust domain, the cloud may take the initiative to disclose the data information to unauthorized parties and may also be invaded to passively disclose user information. The medical cloud server is not trusted, but in order to be able to store patient data securely on it, the data must be encrypted before the medical patient's personal data is outsourced. But then it becomes very difficult for medical data administrators to search for encrypted medical data in a medical server. There are a number of simple ways to do this, such as downloading all the encrypted medical data locally, decrypting all the medical data downloaded, and then searching for keywords in plaintext after decryption, which is simple; but for local storage and network bandwidth, as well as the user's need for decryption and search to pay a huge computational overhead, the operation is extremely inconvenient. The patient or medical users must directly send the required medical file keys and the keywords used to locate the medical file to the medical cloud server to interpret the key, and then based on clear text information on medical data files to search, as we all know, not only the medical cloud server but also the cloud server is not completely trustworthy. By sending the key directly to the medical cloud server, the cloud server will be directly informed of the user's personal information, which is tantamount to the user's personal information published in the network, and there will be no privacy. For the medical cloud, which is now widely used, with the rapid development of its sharing and economy, and not only the medical cloud, but almost all cloud storage systems, like a double-edged sword, where the data stored are mostly related to the privacy of users, if these personal information is leaked, the consequences will affect the lives of these people and even their security, it will be unimaginable. In particular, medical data administrators upload data to the medical cloud, while other medical data administrators, patients, or visitors visit the medical cloud and search for keywords which will give unscrupulous people the opportunity to steal data to raise risk to the safety of medical data. Based on the many keyword searches nowadays, the index structure sk-lsh of Liu

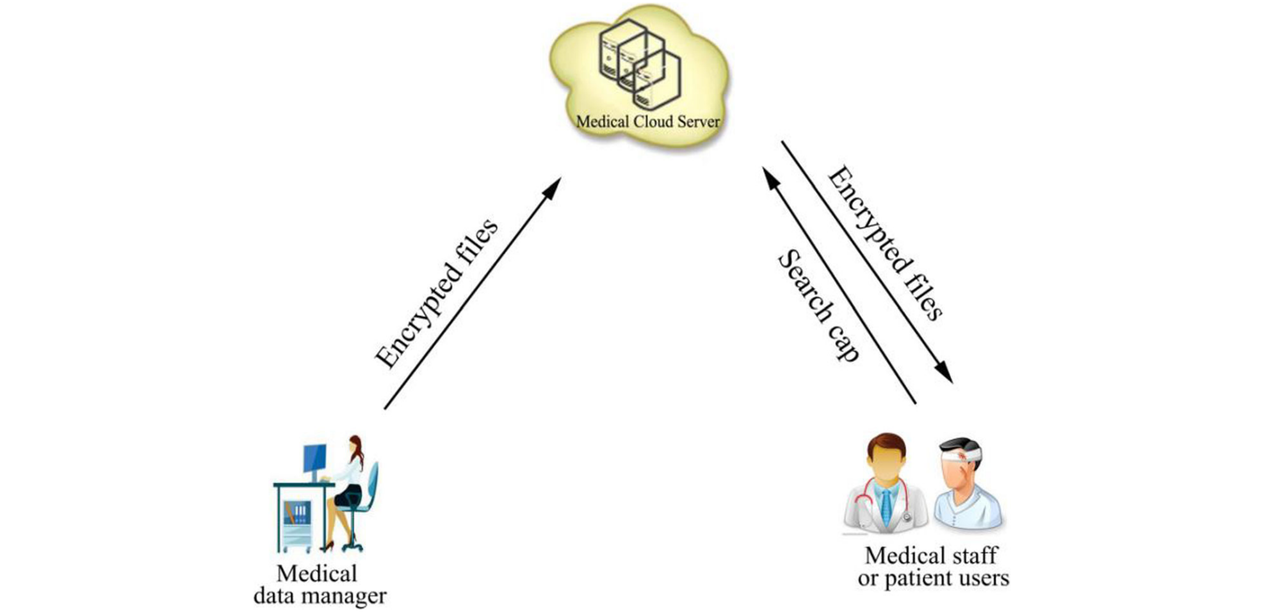

Fig. 1 Users generate search credentials locally

Yingfan and others has ensured the accuracy of the search, but it cannot be applied to the outsourcing ciphertext database scene, which makes it unsuitable for the medical cloud computing environment. Before that, Chai and others put forward a verifiable keyword search scheme. Only the accurate keyword search is supported, and the search scheme of fuzzy keyword based on cipher is proposed by Li et al.; although the search security is ensured to a great extent, it does not support the verifiable problem of keyword search.

Therefore, when used in the medical cloud, the realization of the medical data cannot be decrypted and compromised under the premise that the patient's personal medical data encryption search optimization appears to be particularly important. In order to improve the search performance, the approximate nearest neighbor (ANN) search technique has aroused wide concern in the academia. In order to find a data point close enough to the query data point, the nearest neighbor data point is replaced by the tradeoff between the accuracy of the retrieval results and the retrieval time cost. Previously, Indyk and Motwani proposed the concept of a locally sensitive hash (LSH), which refers to a special hash function with a "margin-preserving" feature that maps a data point close to the same value at a high probability. And conversely, the probability that a data point farther away is mapped to the same value is lower. The locally sensitive hashes are widely used in the field of ANN data retrieval. With the development, LSH is also constantly being improved, and recently, Liu Yingfan and others have designed a new index knot sortingkeys-lsh (SK-LSH), which can effectively reduce the number of page accesses while ensuring the accuracy of the retrieval results. However, the SK-LSH approach requires that users store data indexes locally and cannot apply to outsourced ciphertext database scenarios, which makes them unsuitable for cloud computing environments. Similarly, the medical cloud server is half trustworthy and curious for search operation results returned, but the medical server may only perform part of the search operation and return a part of the content of the results. Chai and other people for the first time, in response to the accuracy of cloud server return results, proposed a verifiable keyword search scheme. Through the method of accurate keyword search and the verification of the result, the previous scheme has improved to a great extent, but it does not support the search method of fuzzy keyword.

Since then, Li et al. have proposed the search scheme of fuzzy keyword based on ciphertext, but it does not support the verifiable problem of keyword search. The key problem is to encrypt the fuzzy keyword search on the medical cloud and to verify the result. This paper presents a verifiable fuzzy keyword symmetric searchable encryption scheme based on medical cloud data outsourcing system, which not only uses symmetric encryption algorithm, but also supports the search of fuzzy keyword, and supports the verification of search results.

This ensures confidentiality of medical data, which is encrypted before the data is outsourced by medical data administrators, and when patients or other users need to retrieve relevant files containing a keyword, they can guarantee their privacy and ensure the integrity, reliability, and security of the returned results. We constructed a system model based on the data relationship between hospital and medical data administrators and patients. Figure 1 shows the system model in which the user generates the search credentials locally, and Fig. 2 shows the system model in which the medical data owner generates the search credentials and sends them to the patient user.

2 Searchable encryption

2.1 Symbol description

Symbol	Definition
ξ	Safety parameters
Key	Secret key
λ	Index
$Y=(Y_1,Y_2,\cdots,Y_n)$	Clear file collection
$W=(W_1,W_2,\cdots,W_n)$	Secret file collection
C	Keywords
$T_C=\text{Trapdoor}(\text{Key},C)$	Keywords trapdoor
$Y(C)=\text{Search}(\lambda,T_C)$	Identifier collection
$Y_i=\text{Decrypt}(\text{Key},W_i)$	Clear text file
Wd	Document collection built on dictionaries
q	Number of queries
$H_q=(Wd,c_1,c_2,\cdots,c_q)$	Query interaction information
I	Document generated index under the key
T_i	Keywords for query history
$V_k(H_q)$	Adversary view of query history
$\alpha(H)=(D(c_1),D(c_2),\cdots,D(c_q))$	Query array of historical visits
Π_q	Matrix of query history patterns
S	Query history query trail
Pr	Probability
$\varsigma(H)=(\vert Y_1\vert,\vert Y_2\vert,\cdots,\vert Y_n\vert,\partial(H),\sigma(H))$	The history of the query history
ϑ	Accurate keyword set
σ	Fuzzy keyword collection
$F_{\sigma'_d}=f(\lambda\tau,\sigma'_d)$	Trap door collection
E_σ	Index tree
A_σ	Medical data file address collection
w_i	Keywords
D_i	File indicator
$D(w_i)$	File identifier set containing keywords

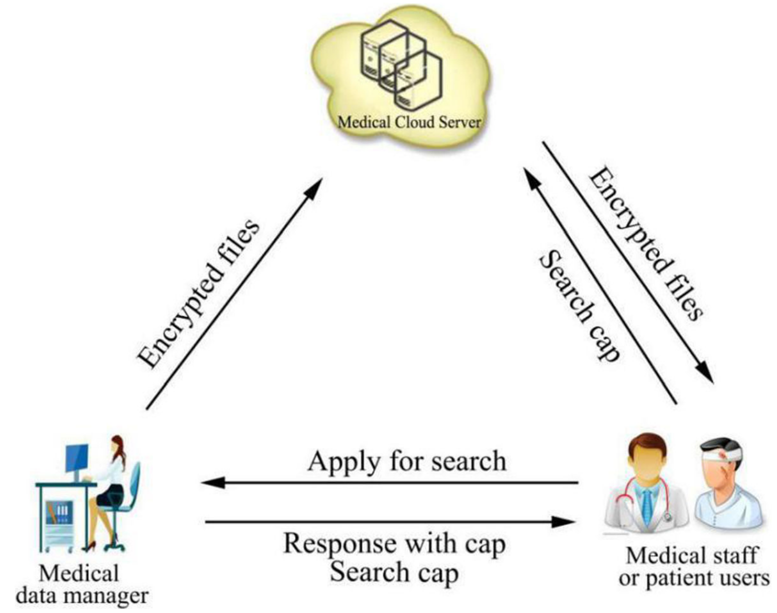

Fig. 2 The medical data owner generates search credentials and sends it to the patient user

2.2 Development of searchable encryption

With the generation and development of cloud storage, searchable encryption can develop rapidly; the research content for searchable encryption is shown in Fig. 3.

But because the time of medical cloud storage technology is short, it has aroused the attention and research of the researchers, thus the application of cloud storage in medical field has led to the development of the technology of searching encryption. The searchable encryption process is shown in Fig. 4.

In the medical aspect, the patient user or the medical data administrator first encrypts the medical data through the searchable encryption mechanism, then stores the medical ciphertext data in the medical cloud server, and when the patient user or the medical data administrator searches for the medical data, the input is the individual keyword. The search voucher for this keyword is sent to the medical cloud server, then the medical cloud will receive a keyword search voucher for each file to try to match; if the match succeeds and the file contains the keyword, then the file is returned to the patient or medical user personnel. The medical data is encrypted before it is uploaded to the medical cloud server, so the patient user or medical data administrator only needs to decrypt the file when receiving medical data returned by the medical cloud. From a security perspective, the medical cloud server does not receive sensitive information about the requested search medical keywords and the medical files within the entire search process. The searchable encryption step is shown in Fig. 5.

This shows that the search encryption mechanism is not only to provide a great convenience for patients or medical personnel and other users and to avoid a large waste of resources, but also to a certain extent to ensure the safety of medical data.

2.3 Symmetric searchable encryption

In 2000, people such as D. X. Song first proposed the concept of searchable cryptography and constructed the first symmetric searchable encryption (SSE) scheme [1]. However, there is no index structure for the file in its scheme, but it is searched by a ciphertext scanning method. When encrypting, the plaintext is segmented into a single word and encrypted using pseudo-random functions and XOR operations. Search, just to retrieve the keywords to the medical server, medical server to all ciphertext to the different or operation, can confirm the existence of keywords. However, it is found that the calculation cost of this kind of scheme increases with the increase of medical data file, which is very limited and inefficient for the great medical data file. At the same time, the medical server can obtain the sensitive information contained in the medical data file by searching the keyword information entered in the medical data file, which poses a great threat to the safety of the patient's personal privacy. To address these deficiencies, E. J. Goh introduced the concept of safe Indexing in 2003 [2]. The

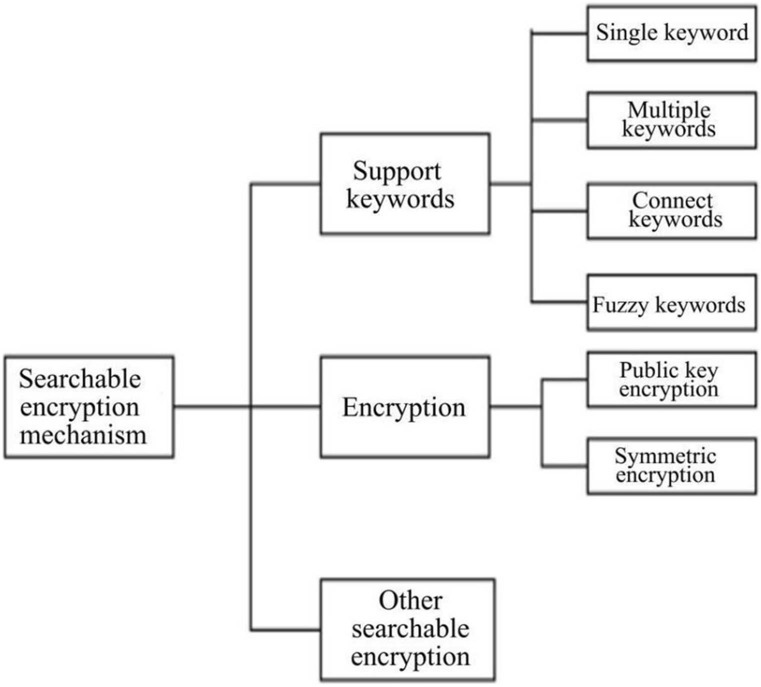

Fig. 3 Searchable encryption mechanism research content

relationship between the number of medical data files and the amount of medical data files is established by establishing a safe index to the collection of files, breaking the cost of the retrieval and the size of the medicine file. When the scheme is established, it is based on the Teflon filter [5], which is the main use of pseudo-random function f by entering the medical data file ID number and searching for medical documents to the keyword and mapping the Prum filter of different signs, for the required medical data files to establish a cloth lung filter. In the search, the user just enters the query keyword, then the retrieval algorithm can retrieve the index of each file. In the Ind-cka security model [2], in the query phase, an attacker can randomly ask for keyword indexes and tokens. In the challenge phase, for two of the same length of files, the challenger randomly selects files and generates an index returned to the opponent, then the adversary cannot distinguish between the index but does not explicitly consider the security of the token. Its security is also an integral part of the searchable encryption mechanism. This is why, in the literature [6], Chang and others put forward the Ind2-cka security target again. Unlike Ind-cka, opponents cannot differentiate between two-length files. E. J. Goh's improved approach can also achieve Ind2-cka security goals by adding redundant data to the index. In the literature [6], the index of each file is indexed by using a dictionary, then the position of each keyword in the index is rearranged by

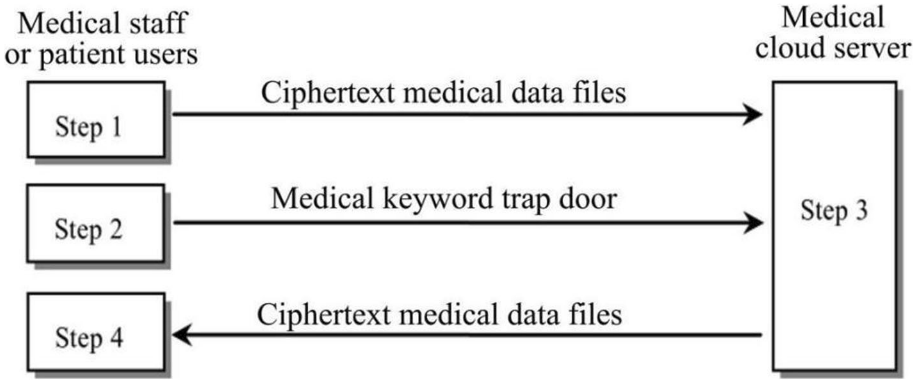
Medical staff or patient users
Medical cloud server
Step 1
Ciphertext medical data files
Step 2
Medical keyword trap door
Step 3
Step 4
Ciphertext medical data files

Fig. 4 Searchable encryption process

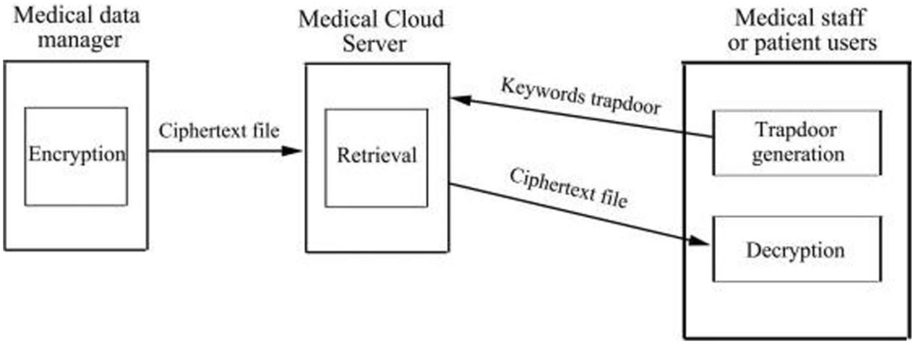

Fig. 5 Searchable encryption step

pseudo-random function, and the index is XOR or manipulated by pseudo-random bit. At the time of retrieval, the medical staff or patient personnel generate tokens to restore the value of the flag bit in the index, to determine whether the file contains the keyword, to ensure the security of the token, and to establish a linear relationship between the cost of medical file retrieval and the number of medical data files, thus making the storage of massive medical data more suitable. The symmetric searchable encryption model is shown in Fig. 6.

In 2006, Curtmola and others pointed out that security tokens in the Chang scheme could be met by any scheme [7] with less security and which then modifies the security definitions for searchable encryption. In the adaptive and non-adaptive attack models, we define the semantic security and the security objective of the indistinguishable security and propose two efficient search encryption schemes. The first scenario uses the inverted index to create a query index for the keywords in the dictionary, storing the file with the specified keyword in the array as a linked list and storing the location and token information of the first node in the quick lookup table that contains the specified keyword list. In the retrieval phase, the medical server uses the retrieval token to restore the location of the first node in the table and then iterates through all the nodes in the key list in the array to find all the files containing the keywords. Because of the structure of the array and linked list, the complexity of retrieval time of medical server is linear with the number of files containing the keyword, and the efficiency is high. However, when updating an existing file, it is expensive to rebuild the index. Therefore, the scheme is suitable for the stability of

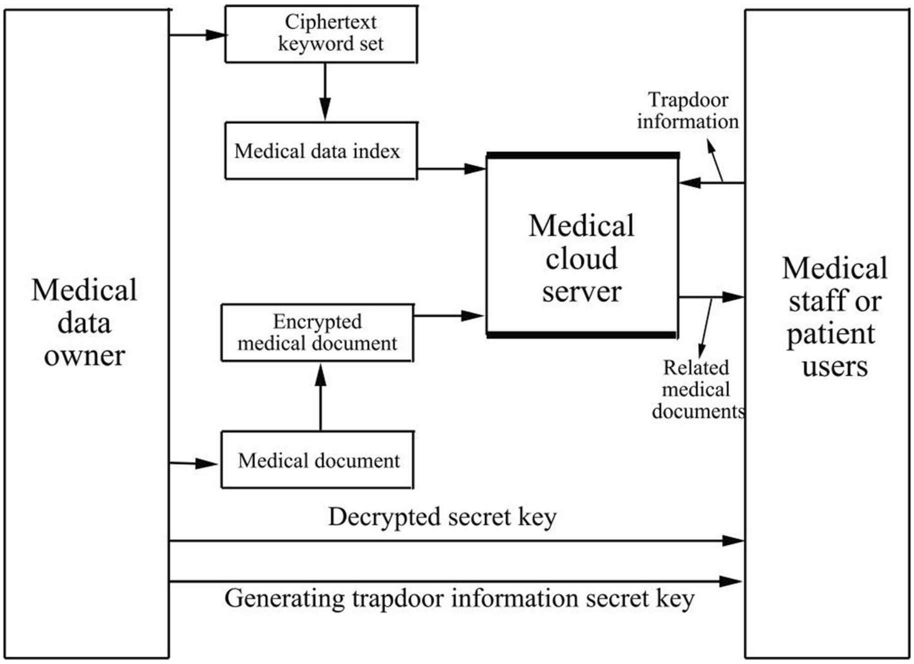

Fig. 6 Symmetric searchable encryption model

the file set; meanwhile, the scheme resisting the choice of keyword attacks is not adaptive safety (security against chosen-keyword attack, CKA1). In the second scenario, instead of using a linked list to store the IDs of all the files containing the keywords, the address fields in the (address, value) two tuples are processed using pseudo-random functions, and the two tuples are stored in the query table, which satisfies the adaptive security definition (adaptive against chosen-keyword attack, CKA2). Finally, a multi-user searchable encryption scheme is designed with the combination of broadcast encryption [8], Curtmola, and a shared secret key is used to achieve multi-user access to encrypted files. Symmetrically searchable encryption steps are shown in Fig. 7.

However, in the medical treatment of the medical staff or patients in the removal of personnel rights, with the need to encrypt the file, the calculation cost is large.

In recent years, the research of a symmetric searchable encryption mechanism mainly focuses on the performance improvement, security optimization, and function expansion of the scheme, and has achieved remarkable achievements.

Because of its retrieval of medical data files, the calculation cost is small, the algorithm involved is simpler, and the operation speed is fast, so it is suitable for the storage of medical data in the medical cloud.

2.4 Symmetric searchable encryption algorithm definition

2.4.1 Algorithm description

Define 1 Symmetric searchable encryption five tuples: (1) Key = KeyGen(ξ), security parameters ξ, output randomly generated key Key, (2) (λ, W) = Encrypt(Key, Y), symmetric key Key and clear file set $Y = (Y_{1,}\ Y_2, \ldots.\ Y_n\ )$, $D_i \in 2^{\Delta}$, index λ and ciphertext file sets $W = (W_1, W_2, \ldots W_n)\lambda = \phi$, (3) T_C = Trapdoor(Key, C), keywords C, traps T_C, (4) $Y(C)$ = Trapdoor(Key, C), identifiers set, (5) Y_i = Decrypt(Key, W_i), clear text files Y_i.

For arbitrarily belonging to the five tuples ξ and n, $C \in \Delta$, $Y = (Y_1, Y_2, \ldots Y_n)$ as well as KeyGen(ξ) and Encrypt(Key, Y), the output's Key and (λ, W) having Search(λ, Trapdoor(Key, C)) = $Y(C)$, and Decrypt(Key, W_i) = Y_i being established.

Define 2 Query history: definition Δ is a dictionary, Wd is a set of documents built on a dictionary.

A history query $H_q \in 2^{2^{\Delta}} \times \Delta^q$ is the interaction information of a client and a medical server $H_q = (Wd, c_1, c_2, \ldots, c_q)$ after q second query.

Given a query history, the adversary obtains all the information in the interaction called the query history view. The view is mainly composed of the index of the document set and the trap gate of the query keyword, as well as some other common information like the number of files in the document set and the ciphertext information. However, the view does not disclose any information except the results and search patterns. Define I_{Wd}, the index Wd that represents a document generated under the key$^{\mathrm{Key}}$, T_i representing the history of the query H_q's keyword trap door.

Define 3 View: definition Wd is a collection of n files on a dictionary Δ, $H_q = (\mathrm{Wd}, c_1, c_2, \ldots, c_q)$ query history after q query, in the security key$^{\mathrm{Key}}$ for the next adversary about H_q the view is:

$$V_k(H_q) = (\mathrm{id}(\mathrm{Wd_1}), \mathrm{id}(\mathrm{Wd_2}), \ldots, \mathrm{id}(\mathrm{Wd_n}), E(\mathrm{Wd_1}), E(\mathrm{Wd_2}), \ldots, E(\mathrm{Wd_n}), I_{\mathrm{Wd}}, T_1, \ldots, T_q) \tag{1}$$

Define 4 Access Mode: The definition Wd is a collection of n files on a dictionary Δ, and the access pattern for H_q querying history after q second query is an array $\alpha(H) = (D(c_1), D(c_2), \ldots, D(\mathrm{n}))$.

Define 5 Query mode: Δ is a dictionary, $\mathrm{Wd} \in 2^{\Delta}$ is a collection of n documents, H_q query after q querying the history of query mode is $H_q = (\mathrm{Wd}, c_1, c_2, \ldots, c_q)$, 1 composition of the symmetric matrix Πq, if $c_i = c_j$, set the matrix of the first $\rangle$ row and $|$ column 1, otherwise set to $0(1 \leq \mathrm{i}, \mathrm{j} \leq \mathrm{q})$.

Define 6 Traces: Δ is a dictionary, $\mathrm{Wd} \in 2^{\Delta}$ is a collection of documents, the query after the q inquiry history is $H_q = (\mathrm{Wd}, c_1, c_2, \ldots, c_q)$, then query history of traces expressed as

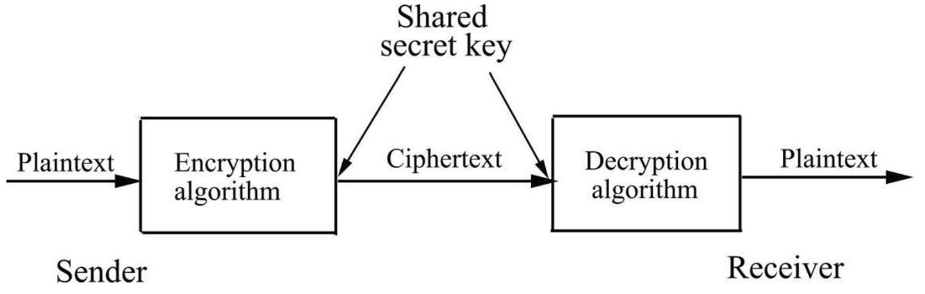

Fig. 7 Symmetric searchable encryption steps

$$\begin{aligned} \mathrm{S}\left(H_q\right) = & (id(D_1), id((D_2)., \ldots, id((D_n), |D_1|, |D_2| \\ & , |D_3|, \ldots, |D_n|, |Dc_1|, |Dc_2|, \ldots, \left|Dc_q\right|, \Pi q). \end{aligned} \tag{2}$$

Define 7 Non-self-adaptive security: If for all$q \in N$, in any probability polynomial time of the adversary $A = (A_1, A_2)$, any polynomial p and large integer ψ, if the following formula is satisfied, it is considered that symmetric searchable encryption is not adaptive and indistinguishable in the sense that it is safe.

$$\begin{aligned} &\Pr\left\{ \begin{array}{c} b' = b : \Psi \leftarrow (1^{\psi});\ \ (H_0,\ \ H_1,\ \ \text{state} \leftarrow A_1) \\ b \leftarrow \{0,1\};\ \ b \leftarrow\ A_2(V_k(H_b),\ \ \text{state}) \end{array} \right\} < \ \frac{1}{2} + \frac{1}{p(k)} \\ &\Pr\left\{ \begin{array}{c} b : \psi \leftarrow\ \left(1^{\Psi}\right);\ \ \ (H_0, H_1, \text{state} \leftarrow A_1) \\ b \leftarrow \{0,1\}; b \leftarrow A_2(V_k(H_b), \text{state}) \end{array} \right. < \frac{1}{2} + \frac{1}{p(k)} \end{aligned} \tag{3}$$

Two query histories and their query traces S are equal; state is a polynomial bounded string used to capture the enemy's state; and Pr is the probability of expression, by the key generation algorithm and the index construction algorithm together to determine.

2.4.2 Security objectives

When designing a cryptographic scheme, the main consideration is the possible security objectives to be met under the attack model, and the security objectives are usually defined in combination with the attack model. As early as 2000, Song [1] introduced the undeniable security objective of the verifiable security theory into a searchable encryption mechanism, requiring the ciphertext to not disclose any original file information. However, Song's original definition is not sufficient to describe the attacker's ability to search in real-world scenarios. In view of this problem, Goh [9] puts forward the Ind-cka security target of choosing keyword attack, which requires an attacker to be able to arbitrarily inquire (or produce in a black box) ciphertext files and keyword traps; there is also no way to get more original file information than through the trap retrieval method. Further, Chang and Mitzenmacher [10] consider the circumstances in which an attacker can obtain a query result from all rounds of the medical server at the time of the attack, and describe the security definition of a searchable encryption mechanism based on impersonation, to limit the medical server's access to any information except for each round of query results. In 2006, Curtmola and other people [11] pointed out that ① Goh [9] did not explicitly consider keyword trap in the search encryption mechanism of security, and in ② Chang Mitzenmacher's study [10], the security definition does not describe an attacker with adaptive attack capability and can be trivially searched by any searchable encryption scheme. Curtmola [11] then formally defines SSE's semantic security (SS) and the indistinguishable safety in adaptive and non-adaptive models (indistinguishability, referred to as IND). Before describing security objectives, several concepts are introduced:

Define:

The hypothesis $\Delta = \{C_1, C_2, \ldots C_d\}$ represents a keyword dictionary, $Y = (Y_1, Y_2, .. \ Y_i)$ that represents a collection of plaintext files, $C = (C^1, C^2, \ldots C^{(q)})$ representing a set of queried keywords, which $Y_i \in 2^{\Delta}$, $C_i \in \Delta$ define the following concepts:

1) q^{-} Inquiry history $H = (Y, C)$, $|C| = q$;
2) H query format $\partial\ (H) = (D(C^{(1)}), D(C^{(2)}), \ldots D(C^{(q)}))$;
3) The search format H isqXq matrix, for $1 \leq i, j \leq q$, if $C^{(i)} = C^{(j)}$, then the i row^{j} column element $\sigma(H)_{ij} = 1$; otherwise, $\sigma(H)_{ij} = 0$;
4) The attacker's H view is defined as:

$$\mathrm{S}_{\mathrm{Key}}(H) = \left(\lambda, W, T_1, T_2, \ldots T_q, id(Y)_1, id(Y)_2, \ldots id(Y)_n\right)) \tag{4}$$

including ciphertext files Key generated under key action and their indexes, traps for historical query keywords, and some additional information, such as each file identifier;

5) H path $\varsigma(H = (|Y_1|, |Y_2|, \ldots |Y_n|, \partial(H), \sigma(H)$ includes H query format, retrieval format, and Y file length information.

The definition of symmetric searchable cryptographic security targets stems from the game process of attackers and challengers: The challenger first executes KeyGen the algorithm to produce the symmetric key Key and, in a certain way, responds to the attacker's query according to the random parameters generated by secret, and finally, by calculating the output of a judgment value as a guess of the random parameter, then the symmetric searchable encryption algorithm is reached, and SSE achieves the corresponding security target.

3 Verifiable search encryption under outsourced data

Medical clouds are not fully trusted, and medical data is stored on the medical cloud, and there is a risk in data retrieval. In order to protect medical data from being compromised and in order to protect the privacy of users, encryption of search data and search verification must be indispensable.

3.1 Verifiable fuzzy keyword search background

In the background of cloud computing, the medical data administrator in the outsourced database model cancels his data center, entrusts the database to the cloud service provider instead of the medical data administrator to manage and maintain the database, and provides the remote database service for the medical data administrator and the medical personnel or the patient personnel. Outsourcing databases [4] have many advantages over local databases; on the one hand, medical data administrators can save the cost of purchasing database hardware and software, hiring professionals to manage the database, and handing the database to cloud service providers for less service costs; on the other hand, because the cloud service provider has the resource concentration advantage, so medical data administrators outsource the database to cloud service providers and can efficiently share resources with other medical personnel or patients. However, for medical data administrators and medical personnel or patients, the local database is trustworthy and the cloud service provider may not be trusted, so the use of outsourced database services will face many security threats, such as data loss, tampering, and leakage.

Shaikh and other security issues such as data loss, data disclosure, authorization of medical personnel or patients, and handling of malicious users are presented by introducing a number of articles related to cloud computing security. There are two core security issues in the outsourced database mode, one is confidentiality and the other is integrity. Confidentiality mainly includes the data confidentiality of medical data administrator and the search privacy of medical personnel or patients, in which data confidentiality means that data outsourced by medical data administrators will not be leaked to unauthorized users [3]. Data disclosure problems can be resolved by means of data encryption.

Search privacy refers to the medical staff or patient personnel's search content will not be leaked to the other side of the interaction, at most to the other side of the disclosure of search patterns and access mode; the two models by Curtmola and others for the first time are introduced. Integrity consists primarily of the storage integrity of the medical data administrator (storage integrity) and the query integrity of the medical or patient staff (query integrity) in two aspects. Storage integrity refers to the tampering and deletion of stored data by a cloud service provider or other attacker that is detected by medical data administrators. Query integrity includes the correctness of search results (correctness, completeness (completeness), and freshness (freshness). Correctness means that the returned search results are not tampered with, completeness means that the returned search results contain all the data that satisfies the search criteria, and freshness refers to the returned search results from the most recently updated database.

Freshness is to the dynamic database; the static database naturally satisfies this nature. To prevent data leaks, people choose to encrypt the data and upload it to the cloud. However, the ciphertext data is different from the clear text data, and its search becomes complicated, so the scholars put forward a variety of searchable encryption models to solve the problem of searching the ciphertext data. Han and others have made a comprehensive classification and summarization of the searchable encryption model and scheme of data on cloud medical server. Because the cloud service provider is not fully trusted, the cloud service provider may retrieve or return a partial source data only for the purpose of saving computational cost and maintaining a good reputation and conceal its dishonesty and medical server anomalies. Therefore, the medical staff or patients with the returned ciphertext data need to verify the integrity of the query.

For query integrity verification issues, Wang and others have summed up the relevant solutions.

There are several ways to verify the integrity of the query, which mainly divided into three categories: the first class is based on the validation data structure (authenticated data structure) method, Devanbu and other people [12] for the first time used Merkle Hashi (MHT) to study the validation of search results, wherein even if a cloud service provider is dishonest, a medical or patient person can verify the correctness of the search results. It (MHT) is a continuation of hash algorithm, which provides effective ways to deal with conflicts in theory and reality. Ma et al. [13] generate a MHT for each tuple, reduce the communication cost of the verification process, and improve the verification efficiency. In addition, M B-free, EM B-gyree, and so on as validation data structures have been presented successively. The second category is based on probabilistic theory, where the Sion allows the data owner to add a forged challenge Token (Challenge token) to the search request, forcing the cloud service provider to perform all search requests; otherwise, the cloud service provider returns the correct Challenge Response (Challenge Response) and the probability will be greatly reduced. Xie [14] let DO add a small number of fake tuples to the medical outsourcing database, and analyze the false tuples in the search results to determine whether the retrieval behavior of cloud service providers is honest; the third kind is based on the digital signature method, Mykletun and others [15] propose a compression RSA digital signature scheme; for a data owner, a plurality of digital signature sets in the search results are used to reduce the computational cost and the communication cost of verifying the correctness of the search results by the medical staff or the patient's personnel. Narasimha [16] proposed a digital signature integration chain (signature aggregation and chaining) scheme, which sorts all tuples by each attribute and then generates a chained digital signature for each tuple based on the order result.

This can effectively validate the integrity of non-empty search results. Wang and so put forward a complete verification of query integrity; the scheme uses the Bloom filter, MHT, and constructed BFT (Bloom filter tree) three data structures, with the help of pseudo-random permutation, hash functions, and other tools, regardless if the return of the cloud service provider search results is not empty. The integrity of the query can be verified by medical or patient personnel. In order to more clearly demonstrate the validation relationship between medical data administrators, medical clouds, and users, we constructed a graphical system model. Figure 8 verifies the search model.

Wang and others put forward a program to support the dynamic updating of the database, which can protect the data privacy of medical data administrators, guarantee the privacy of search for the cloud service providers by medical personnel or patients, and fully realize the verification of the correctness and completeness of the search results by medical personnel or patient personnel.

3.2 Verifiable fuzzy keyword search model

For the fuzzy keyword search method of medical data outsourcing system, we need to construct the medical cloud data outsourcing system, which includes medical data administrator (i.e., data owner), other medical personnel and patients (i.e., users), and medical cloud server. The medical data administrator has a Ω collection of encrypted data δ and a collection of keywords ϑ; the medical cloud can provide the patient or other medical staff with the fuzzy keyword search service in Ω, and medical data administrators and patients have assigned their corresponding permissions, when there is a need for fuzzy keyword search, into the initialization phase, that is, to generate the index generation key$^{\lambda\tau}$ and τ encryption key by manipulating the key generation module on the local server through the medical data administrator, and manipulate the index to establish the module index. For searching for files that contain exact keyword ϑ sets, the patient can operate the trap gate on the local server to generate a trap set $F_{\sigma\prime_d} = f(\lambda\tau, \sigma\prime_d)$ of all the fuzzy keywords σ in the exact keyword set ϑ corresponding to the fuzzy keyword set $\lambda_{\sigma, d}$, and send it to the medical cloud server to launch the search module to perform a search on the index tree E_σ. When the medical cloud receives the search request, the index tree model is shown in Fig. 9.

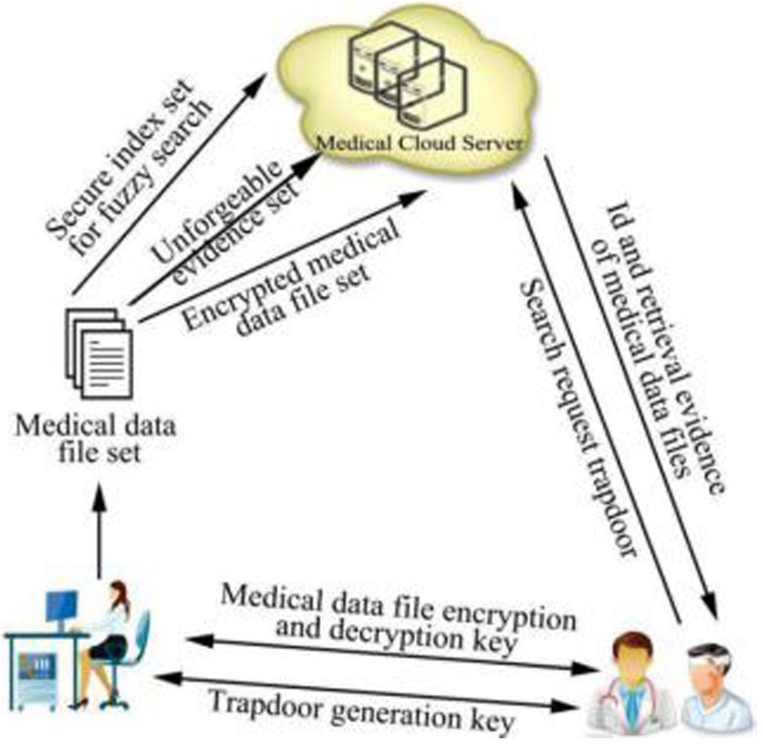
Medical Cloud Server
Secure index set for fuzzy search
Unforgeable evidence set
Encrypted medical data file set
Medical data file set
Search request trapdoor
Id and retrieval evidence of medical data files
Medical data file encryption and decryption key
Trapdoor generation key

Fig. 8 Verification of the search model

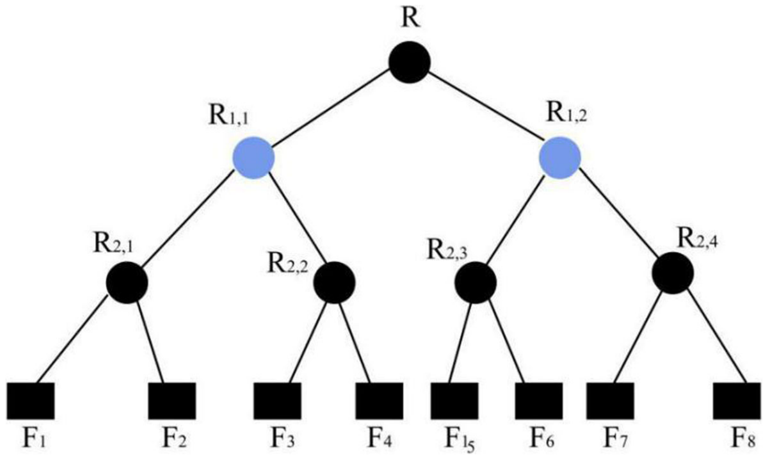

Fig. 9 Index tree model

And it returns the address set $A_\sigma$of all data documents containing all the fuzzy keywords σ and the evidence set Proofset, and the patient can then manipulate the validation module on the local server to verify the integrity of the medical cloud.

Like many cloud servers, medical cloud servers work in a semi-honest mode. The key generation module, the index generation key $\lambda\tau$ and the encryption key τ are made by the random key generation algorithm, and the security parameters used are λ. The medical data administrator uses the index creation algorithm to input the index-generated key $\lambda\tau$ and the exact keyword set Ω of the medical data document collection ϑ, and outputs the index set E_σ. The trap set of all the fuzzy keywords σ corresponding to the exact keyword set ϑ used in the patient search uses the index-generated key $\lambda\tau$ and the fuzzy keyword set $\lambda_{\sigma,\ d}$ as input, and outputs the trap gate set $F_{\sigma_d} = f(\lambda\tau, \sigma_d)$.

The patient operates the input of the authentication module on the local server as the encryption key τ and evidence set Proofset, using the characteristics of pseudo-random function to verify the integrity of the medical cloud server. By constructing the medical cloud data outsourcing system, the medical data administrator operates the key generation module to generate an index generation key $\lambda\tau$ and an encryption key τ, and operation index establishment module creates the index; the patient can operate the trap Gate Generation module, then traps the gate set $F_{\sigma_d} = f(\lambda\tau, \sigma_d)$, and sends it to the medical cloud server. The medical cloud server launches the search module to carry on the search in the index tree and returns the address set $A_\sigma$and evidence set Proofset of the fuzzy keywords σ and validates them. This method not only supports fuzzy keyword search, but also can validate the search results.

A Ω collection of δ encrypted medical data documents is stored on a cloud server by a medical data administrator and an exact set of keywords ϑ. By scanning the encrypted document Ω and then establishing the exact keyword set ϑ, the medical data administrator outsourced the encrypted file to the medical cloud server and got the address of each document A{F_i}, with all the corresponding exact keywords ϑ containing the exact keyword set σ_i. File to write its address was set as $A_{\sigma_i} = A\{F_1\} \mid |A\{F_2\}...| \mid A\{F_i\}$, i for integers greater than or equal to 1, the medical cloud can provide the patient with a fuzzy keyword search service on the encrypted document set Ω and the assigned respective permissions between the medical data administrator and the patient, and the medical cloud data outsourcing system utilizes a symbolic tree-based traversal search scheme. The main idea of its design structure is that all the traps that share a prefix have a common node. The root node is associated with an empty collection, and the symbols in the trap gate can be restored by searching from the root node to the leaf node. All fuzzy keywords can be found in the tree by depth-first search. A set $\Delta = \{\alpha_i\}$ is a predefined set of symbols; there are $|\Delta\ |\ = 2^n$ different number of symbols. And each symbol can be represented by δ bits, δ is an integer greater than or equal to 1.

And each symbol can be represented as a bit, an integer greater than or equal to 1. When a search for a fuzzy-uranium word is required, upon entering the initialization phase, the initialization phase generates an index generation key $\lambda\tau$ and encryption key τ from the key generation module on the local server of the medical data administrator, and the operation index establishment module of the medical data administrator and the patient. Then, in order to search for files containing exact keyword sets ϑ, authorized medical personnel or patients manipulate the trap generation module on the local server to generate all the fuzzy keywords $\lambda_{\sigma,\ d}$ in all the ambiguities in the fuzzy keyword set σ corresponding to the exact keyword set ϑ, while the fuzzy keyword sets edit the distance d according to the preset editing distance for the degree of similarity d between two words, σ_1 and σ_1; the editing distance between them is to convert any one of them to another required operation in a total of three kinds of operations: (1) Substitution: replace one letter in the word with another; (2) Delete: delete a letter from a word; (3) Insert: inserts a letter in the word, edits the distance d the value is an integer.

The first step: Firstly, the fuzzy keyword set $\lambda_{\sigma,\ d}$ storage unit in hospital local server is set to a set of null values.

The second step: When the preset editing distance d of large or equal to 1 o'clock, the hospital local server in the way of decreasing the editing distance d by 1 cycle generation of fuzzy keyword set $\lambda_{\sigma,\ d}$, and the value of the fuzzy keyword set $\lambda_{\sigma,\ d}$ into the corresponding storage unit of the fuzzy keyword set $\lambda_{\sigma,\ d}$, until the value of the edit distance d is 0.

Step three: When the preset edit distance d value is 0 o'clock, then through the verification of query integrity (using plaintext data, additional information, and medical data users can verify the integrity of the query), the local server of the hospital will fill the set of accurate keyword set ϑ of the value directly into the fuzzy keyword set $\lambda_{\sigma,\ d}$ memory unit.

Four steps: When the value of a preset edit distance d is small than 0, hospital local server, the value of the first temporary integer storage variable unit τ is incremented by 1 until the number of elements in the fuzzy keyword set $\lambda_{\sigma,\ d}$ is used as the outer loop body, while the second temporary integer storage variable element j is incremented from the initial value 1 by L until the fuzzy keyword set $\lambda_{\sigma,\ d}$. The number of elements in the value position of the corresponding first temporary integer storage τ variable cell is doubled by 1 as the inner loop body, under the control of the inner circulation body and the outer circulation body, the first temporary integer storage variable unit in the fuzzy keyword set is corresponding to the second temporary integer storage unit J.

When the value is odd, the number of elements in the value position is filled into the third temporary integer storage variable unit temp, and the wildcard * is filled into the fuzzy keyword set $\lambda_{\sigma,\ d}$ corresponding to the first temporary integer storage variable cell τ value position of the corresponding second temporary integer storage variable cell j value plus 1 divided by 2 of the quotient value of the sequence bit appropriate when the value of the second temporary integer storage variable cell j is even; the number of elements in the value position of the corresponding first temporary integral storage variable cell τ in the fuzzy keyword set $\lambda_{\sigma,\ d}$ is filled into the third temporary integer storage variable unit TEMP and the wildcard * is filled into the corresponding first temporary integer storage variable unit in the fuzzy keyword set $\lambda_{\sigma,\ d}$. The value position of the element corresponding to the second temporary integer storage variable cell j is divided by the value of the quotient value of the two sequence positions. This makes use of wildcard technology to establish a fuzzy keyword set.

We use a wildcard * to represent all edits at the same location, and the exact keyword σ_i based on the wildcard character and the edit distance d of the fuzzy keyword set with D is represented as $\lambda_{\sigma_{i,},d} = \{\lambda_{\sigma_{i,},0}, \lambda_{\sigma_{i,},1}, \ldots\lambda_{\sigma_{i,},d}\}$, a set of keywords $\lambda_{\sigma_{i,},d}$ that represent the distance d between the exact keywords σ_i.

In order to realize the fuzzy keyword search, Li et al. uses the edit distance and wildcard characters in the article [11] to construct the fuzzy word set. Editing distances are common mathematical methods used to measure similar strings. Two keywords are known $c_1$and c_2 and their editing distances are $d(c_1,\ c_2)$ represented by c_1 the conversion to c_2 the desired operand. Three basic operands are defined: (1) substitution: replace another character in a word with one character; (2) delete: delete a character in the word; (3) insert: inserts a character into the word. Given a keyword c and edit distance d, $S_{c,\ d}$ to express satisfaction. $d(c,c') \le d$ $d(c,c') \le d$ collection of all c'.

The method of constructing fuzzy sets based on exhaustive is to list all possible variants that satisfy the editing distance in a set. When the edit distance is 1, just add one character. When $d = 1,2$, the number of elements in the fuzzy set is $(2l + 1) \times 26$、$((2l + 1)^2 \times 26^2)$, in which l represents the length of the word. When the length of the keyword increases or the editing distance increases, the number of elements in the fuzzy set increases exponentially, which wastes a lot of storage and computational overhead to store and generate the fuzzy word set.

A trap set $F_{\sigma_d} = f(\lambda\tau, \sigma_d)$ generated using fuzzy keywords σ is sent to a medical cloud server; after receiving a search request from a patient or medical officer for a medical data file, the medical cloud launches the search module on the medical cloud server, performs a search on the index tree E_ϑ, and returns the set of addresses A_σ or evidence set Proofset that contains the medical data file. Finally, authorized patients or medical staff to operate the hospital's local server validation module used to test the integrity of the medical cloud, if through the verification of the search is a success of the fuzzy keyword, or failure.

4 Verifiable symmetric searchable encryption

In traditional searchable encryption schemes, servers are generally considered to be half trustworthy. In this model, the server is faithful and inquisitive, and it enforces the protocol strictly, but tries to find as much secret information as possible from the resources it has, given that the medical server may be selfish in addition to “curiosity” in order to save computing and download bandwidth. And the proposed hash algorithm, the MTH algorithm, is proposed in order to verify the search result (MTH is a continuation algorithm of the hash algorithm, which provides an effective method for dealing with the gap between theory and reality) by generating each tuple MTH, which reduces communication costs and improves verification efficiency. In addition, Chaff and others have proposed a new, stronger server model, called Servers that are half believable and curious. Under this model, the medical server may perform only partial search operations and return partial search results.

4.1 Algorithm description

In order to resist this stronger adversary, this paper proposes a verifiable symmetric search encryption scheme, in which medical personnel or patients can verify the search behavior of the Vsse. The Vsse scheme consists of five algorithms (keygen, pre-process, Querygen, Search, verify), and the keygen algorithm is the key generation algorithm, no longer repeat.

4.1.1 Pre-process

The algorithm is used by medical personnel or patients to create an index λ of the outsourced document Set (:a construction of a word lookup tree), initialized to a full $|\varepsilon|$ fork tree, each node containing a triple group (r_0, r_1, r_2) = (*null*, *null*, *null*) in which r_0 stores the node's literal character, and $r_1$stores the node's prefix signature value.

When traversing the document set and reading each word, the algorithm updates the attribute value of the node, the index tree is established, and in order to protect the plaintext information, the first attribute value of all nodes is deleted. For a node with a property that is still empty, it is filled with random values.

Querygen: This algorithm generates privacy protection requests by medical personnel or patients, using the same approach as indexing. For example, $\pi = (\pi[1], \ldots, \pi[m+1])$, because the hash chain is used, the value of $\pi[i]$ depends on the signature (p [1],…,$p[i-1]$) value of the prefix.

Search The algorithm is performed by the server to find the search request on the index tree and returns the corresponding set of document addresses to the medical personnel or patient personnel. The medical server performs a depth-first traversal of the search request $\pi = (\pi[1], \ldots, \pi[m+1]$)on the index tree λ and the search process for each node r_2 return to medical or patient personnel as evidence and search results.

Verify: By the medical staff or patient personnel to verify the return results of the health server, in reverse order to decrypt the search evidence to verify the results.

4.1.2 Define

Non-interactive verifiable symmetric cryptographic search scheme consists of five polynomial time algorithms: (1) *Keygen*: Generate encryption in the process of the key; (2) *Pre-process*: Build retrieve index and evidence, encrypt index and document set with key, upload to medical server; (3) *Querygen*: A trap that generates a retrieval request by a given key and sends it to the medical server; (4) *Search*: The medical service receives the request from the client, returns the retrieval result of the medical staff or the patient and the evidence of the retrieval process; (5) *Verify*: The client verifies the correctness of the search results based on the results and the retrieved evidence and returns no if yes is returned correctly. In order to more clearly show the correctness of the verified search results, we constructed a graphical security search model. Its safe search model is shown in Fig. 10.

Store a keyword set by using the structure of the word lookup tree, that is, a multi-fork tree on the keyword, the root node is a null character, and each node contains only one character; by the root node to a node through the node of the corresponding string of nodes, the node's child nodes contain different characters. Index build ditto all child nodes of each node have a prefix associated with that node. In the same way that the retrieval request is generated, the medical server takes a depth-first search on the index tree and returns the results of the nodes in the search to the medical or patient personnel as evidence. The medical or patient personnel are then validated against the results and evidence returned. Thus, the validation of the return result is realized.

In 2013, Kurosawa and others proposed an efficient and verifiable symmetric searchable encryption scheme. Record $Y = P\{Y_1, Y_2, \cdots, Y_n\}$ is a set of files, $C = \{c_1, c_2, \cdots, c_m\}$ is a set of keywords, and Index = $\{e_{i,\ j}\}$ is a 0,1 matrix of $m \times n$ to indicate whether the keyword is in file Y_j. $\mathrm{PRF}_{\mathrm{Key}} : \{0,1\}^l \times \{0,1\}^*$ is a pseudo-random function where Key is the key. SKE = (F, J, J^{-1}) is a set of symmetric key encryption scheme, where F is the key generation algorithm, J and J^{-1} are encryption and decryption algorithm respectively. $\mathrm{MAC}_{\mathrm{Key}_m}$ is the label generation algorithm for message authentication, Key_m is the secret key.

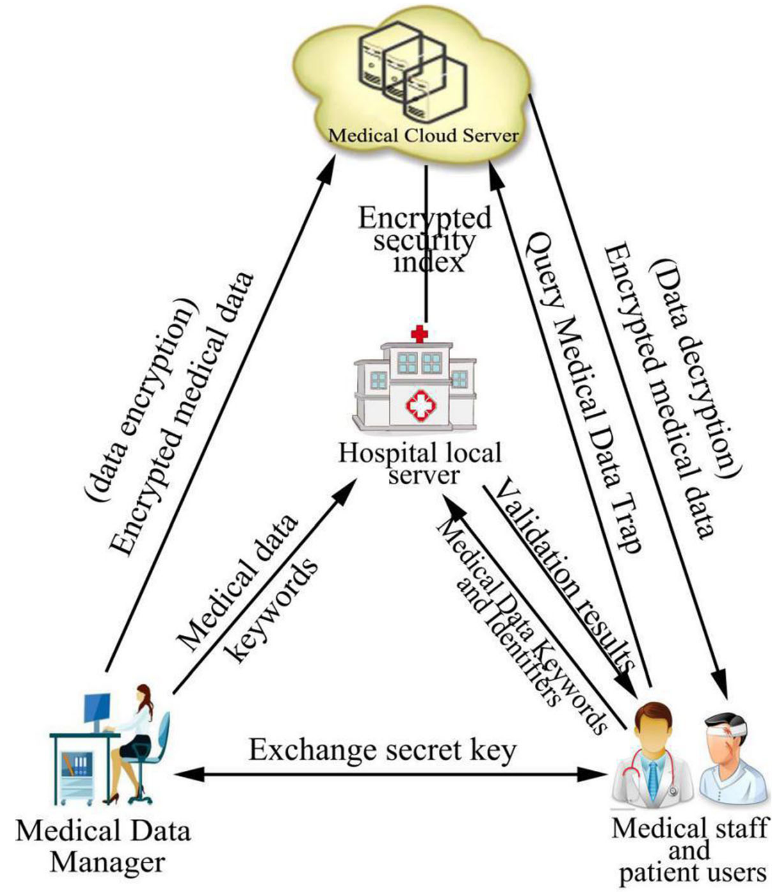

Fig. 10 Secure searchable encryption model

4.1.3 Storage phase

The client generates the key Key_e, Key_9, Key_1, Key_m randomly and then insures the key is secured. Then for each $Y_i \in Y$, $\zeta_i = E_{\text{key}_e}(Y_i)$. For each keyword $c_i \in C$, compute $\text{label}_i = [\text{PRF}_{\text{key}_0}(c_i)]_{1\cdots128}$, $\overline{\text{index}_i} = \text{index}_i \oplus [\text{PRF}_{\text{key}_1}(c_i)]_{1\cdots\cdots n}$, $\text{tag}_i = \text{MAC}_{\text{key}_m}(\text{label}_i, \zeta(c_i))$

,ζ(c_i) is a collection of encrypted files for the keyword c_i. Then send $Y = (Y_1, Y_2, \cdots, Y_n), I = \{(\text{label}_i, \overline{\text{index}_i}, \text{tag}_i) |i = 1, \cdots, \text{m}|\}$ to the cloud server.

4.1.4 Retrieval phase

Set the keywords C_a that the client wants to retrieve, then compute label_a and $\text{pad}_a = [\text{PRF}_{\text{key}_1}(c_a)]_{1\cdots\cdots n}$ send them to the medical server. The medical server uses label_a a stored index set I to match, finds the corresponding index, tag, and calculates $\text{index}_a = \overline{\text{index}_a} \oplus \text{pad}_a$, then gets the index set index_a that contains the keyword c_a, and the medical service returns ζ(c), tag_a.

The client is in the calculation of $\text{tag}_a = \text{MAC}_{\text{key}_m}(\text{label}_a, \zeta(c_a))$, if $\text{tag}_a = \text{tag}_a$. Prove that the returned results are correct, and then decrypt any returned files.

4.2 Medical applications

From Kurosawa and Ohtaki in "secure search symmetric encryption", financial cryptography and data security, a verifiable symmetric searchable encryption scheme (VSSE) is proposed to resist active attack. But we use it flexibly in the medical cloud, then we propose a fuzzy keyword symmetric searchable encryption scheme. The search index of medical data is constructed by inverted matrix to ensure that medical personnel and patients can search the encrypted medical data safely, in which the index of each medical data file has a corresponding keyword. The user or other medical staff can find the index of the required medical data file, that is, the location of the file.

It is assumed that the keywords in dictionary Δ are represented at most d digits. Establishes an index for the document collection D. The index is an array of size 2^d, where Aa | Δ | = m, | D | = n, $id(Di)$ is the file identifier of file Di.

Using pseudo-random permutation π and pseudo-random function f, the parameters are as follows:

$$\begin{aligned} f &: \{0,1\}^k \times \{0,1\}^d \rightarrow \{0,1\}^n \\ \pi &: \{0,1\}^k \times \{0,1\}^d \rightarrow \{0,1\}^d \end{aligned} \tag{5}$$

4.2.1 Specific programs

- Keygen(1^k)

The medical data administrator selects a security parameter k , which can be used to generate the random key $K1$, $K2 \in \{0, , 1\}^k$ with the heap sort, and output the key $K = (K1, K2)$.

- Build Index (K. D)

(1). The medical data administrator scans the entire clear text file collection D, extracts the keyword to generate the dictionary Δ, to each $w_i \in \Delta$, establishes $D(wi)$, $D(wi)$ is a collection of file identifiers containing the keyword wi.

(2). The medical data administrator establishes the index structure based on IM. For each keyword Δin, generates an n-bit index vector Ii, where each $I_i[j]$ represents whether the corresponding file D_j contains w_i. For each file $D_j(1 \leq j \leq n)$, if $D_j \in D(w_i)$, place $I_i[j]$, otherwise $I_i\{j\} = 0$.

(3). Make I an array of size 2^d. For each keyword $w_i \in \Delta$ medical data administrator calculates $\pi_{K_i}(w_i)$ and then places the other $2^d - m$ entries in the $I[\pi_{k_i}(w_i)] = I$ to n-bit 0 vectors.

(4). For all addr = 1, …2^d, compute $f_{K_2}(addr)$.To i = 1, …, m, if the addr satisfies $\mathrm{addr} = \pi_{K_i}(w_i)$, then place $I(\mathrm{addr}) = I_i \oplus f_{K_2}(\mathrm{addr})$; otherwise, place $\mathrm{I}(\mathrm{addr}) = f_{K_2}(\mathrm{addr})$.

(5). Medical Data Manager Output Index I。.

- Trapdoor (k, w)

Enter the keyword to search w, medical data administrator to calculate $\pi_{K_i}(w)$ and $f_{K_2}(\pi_{k_i}(w))$ output search trap door

$$T_w = \left((\pi_{K_i}(W)), f_{K_2}(\pi_{k_i}(W))\right) \tag{6}$$

- Search (I, T_w)

(1). Resolves a trap gate T_w to (α, β). Medical Cloud Server received T_w, first by α positioning to the corresponding index vector, recorded as γ, using β decryption γ, make $\theta = \gamma \oplus \beta$. If the description does not contain the keyword to be queried, the medical cloud server feeds a failure message to a medical or patient person.

(2). Otherwise, the medical cloud server scans every bit of the index vectors. Order is an empty set, yes, if, will join. Finally, the ciphertext corresponding to each file identifier is returned to the medical personnel or the patient personnel. Our medical data on the medical cloud were stored in the storage phase

$$I(\mathrm{addr}_{\mathrm{Bob},1}) = 3, \mathrm{I}(\mathrm{addr}_{\mathrm{Bob},2}) = 6 \tag{7}$$

$$I(\mathrm{addr}_{\mathrm{Bob},3}) = 10, .. \tag{8}$$

$$I(\mathrm{addr}_{\mathrm{Bob},1}) = N \tag{9}$$

To achieve reliability, add a Mac as tag in the index .

During the search phase, the patient’s data is sent to the medical server, the medical server carries out the search return results to the patient, and the patient examines the validity of each tag. Finally, when the patient inquires the data, it gets the corresponding clear text and then gets the corresponding data to be queried. Message verification is shown in Fig. 11. Through the proposed verifiable fuzzy keyword searchable encryption scheme, the first level is inspired by the Kurosawa method, and the verification problem of return result is solved.

$$\begin{gathered} i = 1, 2, 3 \\ \mathrm{tag}_i = \mathrm{MAC}_K\left(\mathrm{addr}_{\mathrm{Bob},1'} C_{\mathrm{Bob},j}\right) \end{gathered} \tag{10}$$

$$\begin{gathered} 1 = 4, ..., N \\ \mathrm{tag}_i = \mathrm{MAC}_K\left(\mathrm{addr}_{\mathrm{Bob},1'} C_{\mathrm{Bob},j}\right) \end{gathered} \tag{11}$$

5 Experimental method

5.1 Medical data security

In order to ensure the safety of medical data when medical data is stored on the medical cloud and when patients or other medical users retrieve medical data from the medical cloud, we outsource the medical data

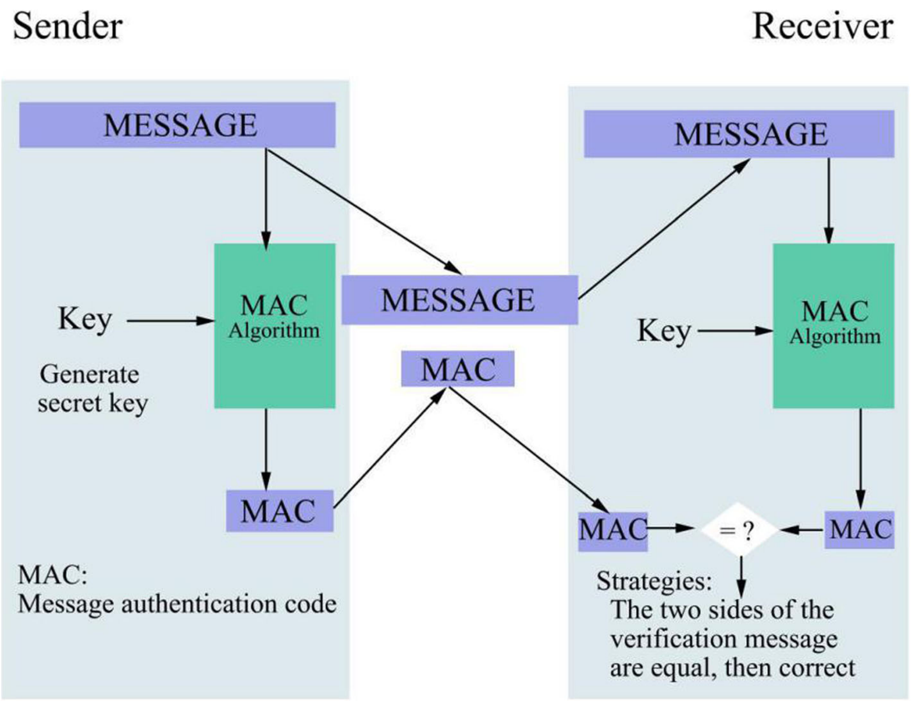

Fig. 11 Message verification

to the medical cloud through the data outsourcing system, in order to protect the user's personal privacy. For security, medical data administrators must first encrypt the medical outsourcing data to the medical cloud.

5.1.1 Experimental method

After the medical data administrator uploads the encrypted medical outsourcing data to the medical cloud, the data user first executes the $\text{Key} = \text{KeyGen}(\xi)$ algorithm to generate the random key Key, and generates the symmetric key and the ciphertext by using the random key and the plaintext file set $Y = (Y_{1,}\ Y_2, \ldots.\ Y_n\ )$, $D_i \in 2^{\Delta}$ through the $(\lambda, W) = \text{Encrypt}(\text{Key}, Y)$ algorithm. The file set $W = (W_1, W_2, \ldots W_n)\lambda = \phi$, trapdoor $T_C = \text{Trapdoor}(\text{Key}, \text{C})$, and finally get the plaintext file through $Y_i = \text{Decrypt}(\text{Key}, W_i)$.

5.2 Medical data integrity

Integrity mainly includes the storage integrity of the medical data administrator and the query integrity of the medical staff or patient personnel. Storage integrity means that the tampering, additions, and deletions of stored data by cloud service providers or other attackers are perceived by medical data administrators. Query integrity includes the correctness, completeness, and freshness of search results.

5.2.1 Experimental method

The index generation key$^{\lambda\tau}$ and τ encryption key are generated by manipulating the key generation module on the local server through the medical data administrator and manipulating the index to establish the module index. For searching for files that contain exact keyword ϑ sets, the patient can operate the trap gate on the local server to generate a trap set $F_{\sigma\prime_d} = f(\lambda\tau, \sigma\prime_d)$ of all the fuzzy keywords σ in the exact keyword set ϑ corresponding to the fuzzy keyword set $\lambda_{\sigma,\ d}$, and send it to the medical cloud server to launch the search module to perform a search on the index tree E_{σ}. And returns the address set $A_{\sigma}$of all data documents containing all the fuzzy keywords σ and the evidence set Proofset, and the patient can then manipulate the validation module on the local server to verify the integrity of the medical cloud.

6 Results and discussion

6.1 Results analysis

In order to ensure the security of medical data, we proposed the above solution and proved the feasibility of our solution through a large amount of simulation data.

Figure 12 shows the trend of creating index time with the number of keywords. The more keywords, the longer it takes to create an index. Note that the index is created only once by the data owner, and the query operation is performed more frequently; because the

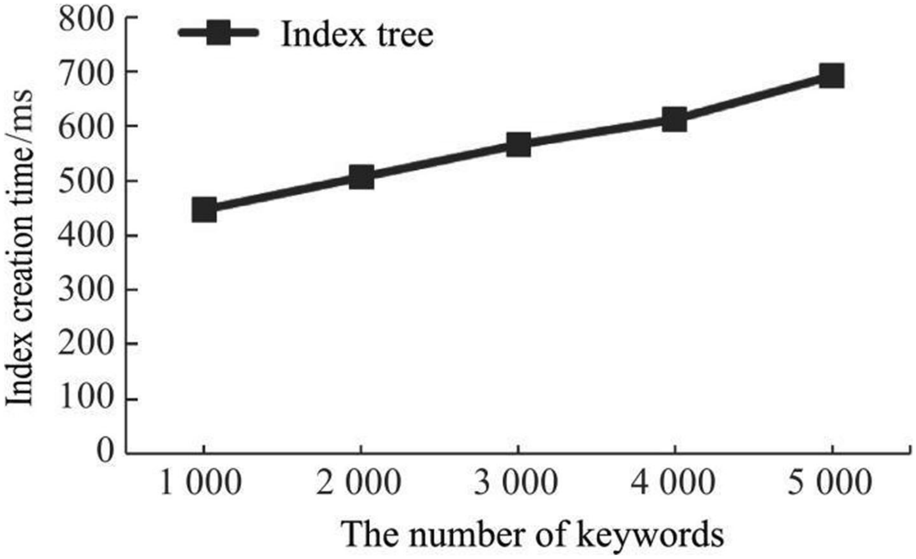

Fig. 12 Time of building index. ■ shows the change in the index creation time as the number of keywords increases

index tree can greatly improve the query efficiency, it is more conducive to improving the overall efficiency of the system.

Figure 13 shows the trend of the creation time of the verification information with the number of keywords. Among them, the verification information refers to the relevant data that needs to be verified during the verification phase. The more keywords, the longer the verification information is created. The verification information is the same as the index structure and only needs to be created once.

Figure 14 shows the change trend of the time of encrypted query with the number of query keywords. The more query words, the larger the semantic expansion set, so the more time the query is encrypted.

Figure 15 shows the increase in the query time with the number of keywords. With the increase of the number of keywords, the query time based on the inverted index increases more obviously, and the query time based on the symbol index tree does not increase significantly. The query time based on the symbolic index tree is obviously less than the inverted index. This is because the symbolic index tree merges the hash segments with the same prefix into one node, which can eliminate irrelevant information more efficiently and greatly improve the query efficiency.

Figure 16 shows the number of different query terms, and the verification time varies with the number of keywords. With the increase in the number of keywords, the number of verification data returned by the medical cloud server is also greater, that is, the user needs to verify the accuracy and completeness of more indexes and ciphertexts, and therefore, the more time is consumed.

6.2 Risk analysis

In the process of collecting and using experimental data, some potential errors and risks will affect the experimental results. Data acquisition is expected to be a true reflection of the current situation completely true data, but it is known that this is not possible. Based on the statistical analysis of the collected data, a reasonable confidence interval is selected to analyze the absolute risk and relative risk at a confidence level of not less than 95%. Eliminating the absolute risk of data bias and human error caused by different patients in medical data has very little impact on our experimental results.

6.3 Discussion

In our scheme, there are still some problems to be solved. In the face of the rapid development of cloud storage, as well as massive medical data, the development of medical cloud storage is unstoppable; in order to ensure the safety of personal information of patients, the convenience of medical cloud storage and the requirement of confidentiality are more and more high. In our scheme, based on the searchable encryption of symmetric encryption algorithms, the encrypting party and the decrypting party must implement the key negotiation in advance, and the key needs to be transmitted through the secure channel, thereby causing unnecessary troubles. The fuzzy search method, which searches through many data in the medical cloud, cannot accurately locate the medical data files that we

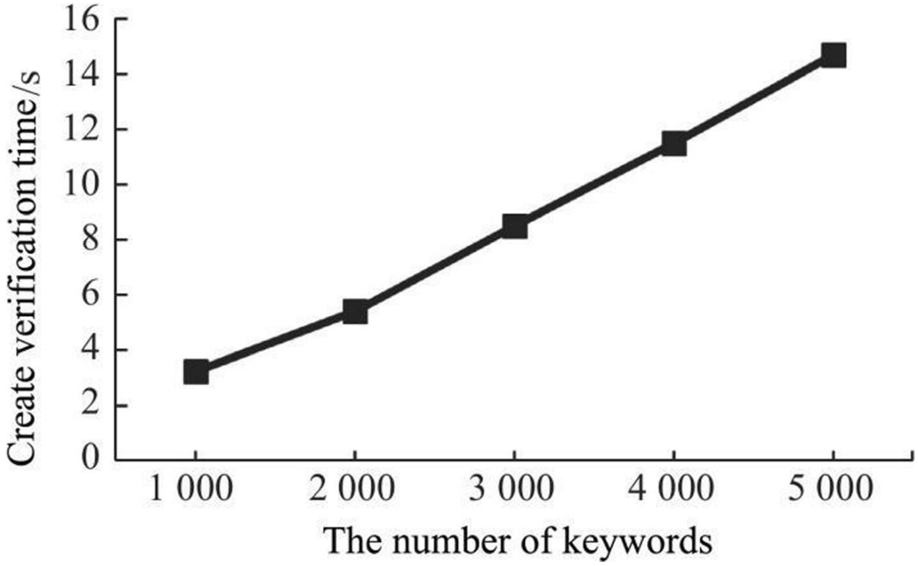

Fig. 13 Time of building verification information. ━■━ shows the trend of creating verification time as the number of keywords increases

need. It will result in time consumption in the medical cloud and filtering.

7 Security analysis

In this scenario, the following process happens: (1) The document: The data owner encrypts the plaintext medical data file using a symmetric encryption method to secure the privacy of the medical data before outsourcing the file, and sends the key to the authorized patient user via the secure channel. Without a key, it is difficult for the medical cloud server or attacker to decrypt the ciphertext document. (2) Index: In each index, the keyed hash function guarantees the privacy of the keyword; the medical data file identifier is the indicator of the associated ciphertext document and does not reveal related information. Therefore, the privacy of the index can be guaranteed. (3) Query: In trapdoors, each query keyword is encrypted by a keyed hash function, so its privacy security is guaranteed. (4) Related verification information: This scheme inserts encrypted keywords in the Bloom filter. The medical cloud server or attacker cannot know the relevant information of the hash function without a key and cannot analyze the Bloom filter. Therefore, the Bloom

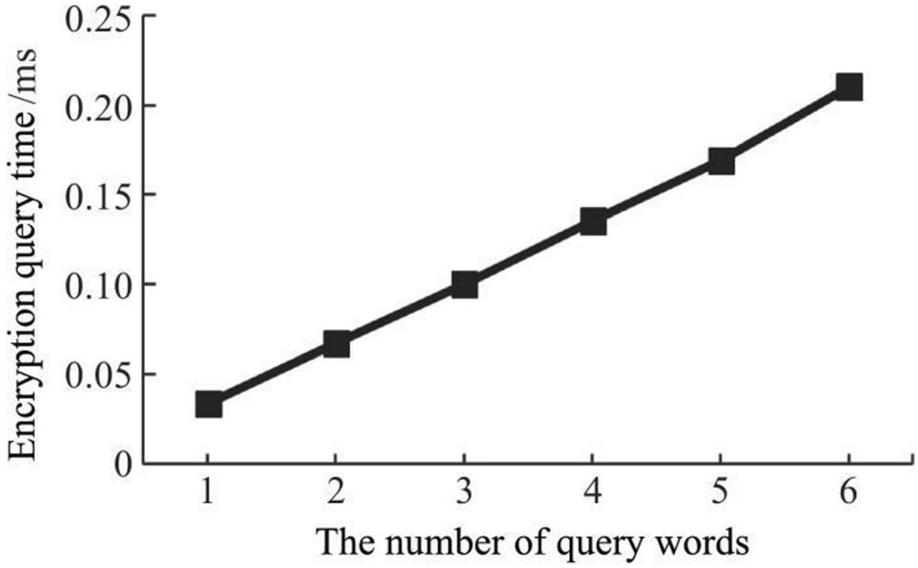

Fig. 14 Time of encrypting query. ━■━ shows the use of the number of query keywords to show the trend of the encryption query time

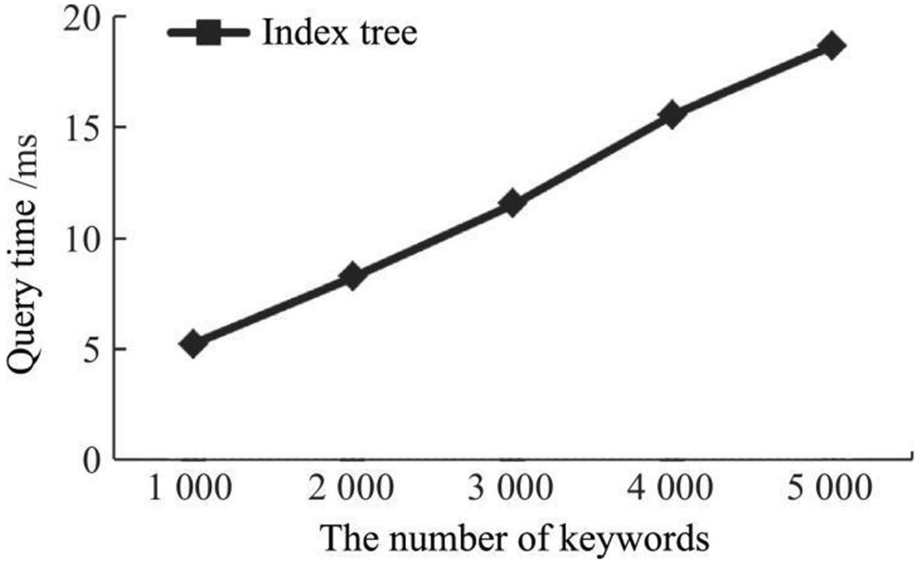

Fig. 15 Time of searching. ■ shows the increase in the query time with the number of keywords

filter is privacy-safe. The message authentication code is obtained through the function MAC, and only the user with the key can use it to verify the result.

8 Conclusion

Although there are still some problems, the algorithms mentioned in our solution have great value for the storage of medical clouds and medical data today. Through the medical data obtained by the technology contained in the intelligent network, the optimal application scheme of fuzzy searchable encryption for the symmetric verifiable fuzzy medical outsourcing system is studied through the medical data obtained. In the process of the medical cloud server, based on the symmetric encryption algorithm, the medical data administrator uses the encryption algorithm to encrypt the medical data and encryption key, and then sends out the data after the encrypted medical outsourcing. After the medical personnel or patient user receives the ciphertext, the decryption method needs to use the

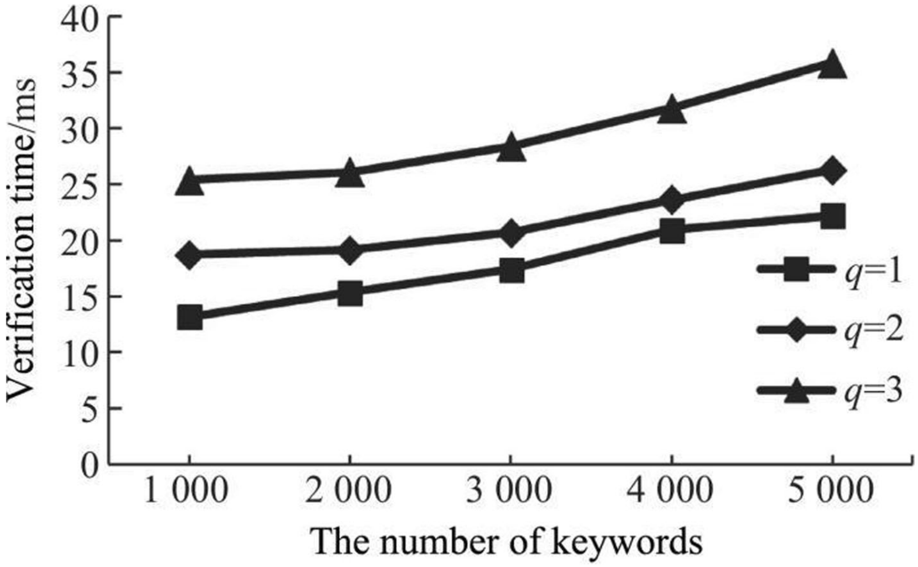

Fig. 16 Time of verifying. ■ shows the change curve between the verification time and the number of keywords when the number of query words is 1. ◆ shows the change curve between the verification time and the number of keywords when the number of query words is 2. ▲ shows the change curve between the verification time and the number of keywords when the number of query words is 3

encryption key used and the inverse algorithm of the same algorithm, then it can interpret the encrypted ciphertext data as plaintext. Symmetric encryption algorithm can be used to realize the public, in the access to the medical cloud server to perform operations when the calculation is small, encryption and decryption are faster, and encryption efficiency is high. But for the verifiable algorithm, it can protect the data file of the hospital and the safety of the patient's medical data to the greatest extent, and the fuzzy search facilitates the search of medical data by the medical staff and patients.

Abbreviations

ANN: Approximate nearest neighbor; CKA1: Security against chosen-keyword attack; CKA2: Adaptive against chosen-keyword attack;
IND: Indistinguishability; LSH: Locally sensitive hash; MHT: Merkle Hashi;
SSE: Symmetric searchable encryption

Acknowledgements

This work was supported by the Natural Science Foundation of Shandong Province under Grant No.ZR2017MF029.

Funding

The research is funded by the Natural Science Foundation of Shandong Province.

Availability of data and materials

In this paper, the medical data used by the Institute come from open source medical data, which can be accessed to see all the data used by the Institute, which is https://github.com/beamandrew/medical-data. The site is a Med Pix database made up of nearly 60,000 medical images from hundreds of thousands of medical patients, which contains statistics on medical data for patients with various diseases, and provides more detailed medical data for our medical cloud storage research. These medical data are classified according to different types of diseases, which can provide clear and detailed data for medical research. The medical data website is an open source website, which can directly access medical data and visit and use it directly through the website.
We, according to the specific medical data, show on the image, by using the method we studied, very clear the impact of medical data cloud storage.

Authors' contributions

HZ and QC contributed to the conception and algorithm design of the study. YW and MS contributed to the acquisition of simulation. HZ and YW contributed to the analysis of simulation data and approved the final manuscript.

Author's information

Huiqi Zhao received his B.Sc and M.Sc degree from Shandong University of Science and Technology, China, in 2003 and 2009 respectively. He is currently working toward his Ph.D degree in Shandong University of Science and Technology. He is a Ph.D and lecturer in the Department of Information Engineering, Shandong University of Science and Technology, Taian, Shandong Province and Qilu University of Technology (Shandong Academy of Sciences), Shandong Computer Science Center(National Supercomputer Center in Jinan), Shandong Provincial Key Laboratory of Computer Networks, Shandong Jinan. His research interests are computer networks, wireless body area networks and healthcare applications, and data privacy protection. (E-mail:zhqskd@163.com)
Qian Chen is a college student in the Information Engineering Department of Shandong University of Science and Technology. He is interested in cloud storage and network information security. (Email: 18853851251@163.com)
Yinglong Wang is a doctor and researcher in Qilu University of Technology (Shandong Academy of Sciences), Shandong Computer Science Center (National Supercomputer Center in Jinan), and Shandong Provincial Key Laboratory of Computer Networks, Jinan, Shandong Province. His research interests are Information security, computer forensics, and wireless sensor network technology. (E-mail:wangylscsc@126.com)
Minglei Shu is a doctor and associate researcher in Qilu University of Technology (Shandong Academy of Sciences), Shandong Computer Science Center (National Supercomputer Center in Jinan), Shandong Provincial Key Laboratory of Computer Networks, Jinan, Shandong Province. His research interests are wireless sensor network, wireless body domain network, and cloud health. (E-mail:smlsmil1624@163.com)

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

[1]Shandong Province Key Laboratory of Wisdom Mine Information Technology, Shandong University of Science and Technology, Qingdao 266590, China. [2]Shandong Provincial Key Laboratory of Computer Networks, Qilu University of Technology (Shandong Academy of Sciences), Shandong Computer Science Center (National Supercomputer Center in Jinan), Shandong Jinan 2750014, China.

Received: 22 August 2018 Accepted: 7 November 2018
Published online: 06 December 2018

References

1. D.X. Song, D. Wagner, A. Perrig, *Practical Techniques for Searches on Encrypted. Data[C]//In Proceedings of the 2000 IEEE Symposium on Security and Privacy* (IEEE Press, Berkeley, 2000), pp. 44–55.
2. E.J. Goh, *Secure indexes[J]. IACR Cryptology ePrint Archive, 2003* (2003), p. 216.
3. R. Agrawal, J. Kiernan, R. Srikant, et al., *Order-preserving encryption for numeric data[C]//Proceedings of the 2004 ACM SIGMOD International Conference on Management of data* (ACM, 2004), pp. 563–574.
4. Ted Pedersen.Information content computed on various corpora [EB/OL]. (2013-5-19) [2016-08-15]. http://wn-similarity.sourceforge.net/.
5. B.H. Bloom, Space/time trade-offs in hash coding with allowable errors[J]. Commun. ACM **13**(7), 422–426 (1970).
6. C. Yancheng, M. Mitzenmacher, *Privacy Preserving Keyword Searches on Remote Encrypted Data[C]//Applied Cryptograpy and Network Security* (Springer Press, New York, 2005), pp. 442–455.
7. R. Curtmola, J. Garay, S. Kamara, et al., *Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions[C]//in Proceedings of the 13th ACM Conference on Computer and Communications Security* (ACM Press, Alexandria, 2006), pp. 79–88.
8. A. Fiat, M. Naor, *Broadcast Encryption[C]//Advances in Cryptology-CRYPTP'93* (Springer Press, Santa Barbara, 1993), pp. 480–491.
9. Goh E. Secure indexes. Technical Report, 2003/216, IACR ePrint Cryptography Archive, 2003. http://www.oalib.com/references/7493839.
10. Y. Chang, M. Mitzenmacher, in *Proc. of the Applied Cryptography and Network Security. LNCS 3531*, ed. by J. Ioannidis, A. Keromytis, M. Yung. Privacy preserving keyword searches on remote encrypted data (Springer-Verlag, Berlin, 2004), pp. 391–421. https://doi.org/10.1007/11496137_30.
11. R. Curtmola, J. Garay, S. Kamara, R. Ostrovsky, in *Proc. of the 13th ACM Conf. on Computer and Communications Security (CCS 2006)*. Searchable symmetric encryption: Improved definitions and efficient constructions (ACM Press, New York, 2006), pp. 79–88.
12. P Devanbu, M. Gertz, C. Martel, et al. Authentic data publication over the internet[J]. Journal of Computer Security.
13. MA Di, DENG R H, PANG H, etal. Authenticating query results in data publishing[C]// Proceedings of the International Conference on Information and Communications Security.
14. XIE Min, WANG Haixun, YIN Jian, et al. Integrity auditing of outsourced data[C]// Proceeding of the 33rd International Conference on Very Large Data Bases.
15. E. Mykletun, M. Narasimha, G. Tsudik. Authentication and integrity in outsourced databases[J], ACM Transactions on Storage.
16. M. Narasimha, G. Tsudik. DSAC: integrity for outsourced databases with signature aggregation and chaining[C]// ACM CIKM International Conference on Information and Knowledge Management.