INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 8, Issue 2 - V8I2-1146)
Available online at: https://www.ijariit.com

Enabling authorized encrypted search for multiauthority database

Kosuru Sneha
snehakosuru07@gmail.com
Dr. M.G.R. Educational and Research Institute, Chennai, Tamil Nadu

Mallina Alekhya
alekhyamallina249@gmail.com
Dr. M.G.R. Educational and Research Institute, Chennai, Tamil Nadu

G. Pooja
poojagajenthiran@gmail.com
Dr. M.G.R. Educational and Research Institute, Chennai, Tamil Nadu

T. Kiruba Devi
kirubadevi.t@drmgrdu.ac.in
Dr. M.G.R. Educational and Research Institute, Chennai, Tamil Nadu

Dr. T. V. Ananthan
tvananthan@drmgrdu.ac.in
Dr. M.G.R. Educational and Research Institute, Chennai, Tamil Nadu

K. Lokesh
koti.lokesh33@gmail.com
Vel Tech University, Chennai, Tamil Nadu

ABSTRACT

E-medical records are sensitive and should be stored in a medical database in encrypted form. However, simply encrypting these records will eliminate data utility and interoperability of the existing medical database system because encrypted records are no longer searchable. Moreover, multiple authorities could be involved in controlling and sharing the private medical records of clients. However, authorizing different clients to search and access records originating from multiple authorities in a secure and scalable manner is a nontrivial matter. To address the above issues, we propose an authorized searchable encryption scheme under a multi-authority setting. Specifically, our proposed scheme leverages the RSA function to enable each authority to limit the search capability of different clients based on clients' privileges. To improve scalability, we utilize multi-authority attribute-based encryption to allow the authorization process to be performed only once even over policies from multiple authorities. We conduct rigorous security and cost analysis, and perform experimental evaluations to demonstrate that the proposed scheme introduces moderate overhead to existing searchable encryption schemes. Index Terms—Multi-authority, encrypted data search, e-medical system, cloud storage, forward security.

Keywords: *Multi-Authority, Encrypted Data Search, E-Medical System, Cloud Storage, Forward Security*

1. INTRODUCTION

Digitalized medical documentation plays a crucial role in this and upcoming digital world of sectors like food, medical, transportation, imports and exports. Which is used to create, manage, and maintain the record. Especially in health sector. Encryption of data before uploading to the cloud .so, that only authorized client who has the key can only decrypt and permit to access. And also, data owner also need key to access their health record. These all encryption and decryption leads to the huge computation and communication cost. And also, it is not possible with the single authority and its only possible by multiple authority through various keywords. To enable search of encryption data we proposed data with client secure search token. As the medical data are sensitive. it should be handled so delicately. So, a patient takes treatment in different hospitals with various doctors. So, all doctors cannot access the entire details of the patient. if the patient consults a neurologist. the doctor can only access the details only related to nerves and other than this he cannot access.

As medical data is totally a privacy thing. It can be access through only authorized keywords for searching. We proposed RSA functionality to search token. ABE it has certain policies, only if it satisfies these policies, it can decrypt and access data . RSA-based Access-Tree CP-ABE scheme that is considered as an efficient and lightweight encryption system that can run on resource constraint easy to use Typically, there are four main algorithms of a CP-ABE scheme, which respectively are setup, key generation, encryption and decryption, CP-ABE use tree structure with different keys into order to Access given attributes.

2. RELATED WORK

Data Encryption and Search

The searchable encryption scheme was first proposed by Chor et al in 1995 and it is based on symmetric encryption. Encryption is the process of encoding information. The process means which converts plain text into a ciphertext where the users or clients who are authorized can decipher a ciphertext back to plaintext and access information. There is also asymmetric encryption which is also

known as public key encryption .it works on two keys where one is Used for encryption and different key is used for decryption. That different encryption key is shared publicly for anyone to use. Currently, many researchers are focused in exploring practical and secure data for large encrypted data search. Many researchers are in searching of a perfect scheme then cast et al designed a novel encrypted scheme that supports Boolean queries and this work is about to realize sublinear conjunctive search for various structured data. Security of data is always a primary concern for researchers or users. Thinking of forward security in cryptography primitive which has proposed to prevent matching the token with documents to server from query.

Attribute Based Encryption

In an ABE system, an encrypted data is associated with a set of attributes. A user's private key is associated with an access structure over a set of attributes. The user's private key reflects the user's access policy. This implies that the user is allowed to decrypt if and only if the set of attributes of a user's private key satisfies the access policy. Comparing with the original ABE scheme, the advanced ABE scheme improved it's expressibility, which means that the user's private key is able to express any monotone access formula consisting of AND, OR, or threshold gates. Moreover, when encrypting a message, a user may not be aware of the attributes. After creating the cipher-text, a new set of attributes may be used in the system. The core component of the current ABE systems is a secret-sharing scheme. The ABE algorithm can encrypt the text from a data owner which correlates the attributes of the user. The main characteristic of ABE is to reconsider the concept of a public key. Normally, a receiver decodes an encrypted message with a public key. In identity-based encryption cryptography, the user's public key can be any string such as an email address. Thus, the cipher-text can be decoded only if someone holds the key with the matching attributes. Generally, the user's key is issued by a third trusted party. ABE is basically a one-to-many algorithm that sends a message so that all legitimate users are able to decode. By contrast, a one-to-one encryption algorithm has scalability issue as it can only send the message to a single recipient.

RSA

Rivest-Shamir-Adleman (RSA) is an asymmetric algorithm that uses key pairs (public keys & private keys) to deal with the message. Public keys can be exposed to anyone. On the contrary, only the owner of keys holds the private keys. In other words, there is no need to compromise security of public keys which can be distributed publicly, but the privacy of private keys. In order to generate the key pairs safely, cryptographic algorithms based on mathematical problems to generate one-way functions are needed. It can be a notion of a trapdoor function which is a mathematical function that underpins the public key encryption system. For example, the process of taking a given value A and using the trapdoor function to get another value B is very easy, however, it is intractable to use trapdoor function to get the value A from the value B. The reason is that it is easy to "add" points together and to "multiply" a point by an integer by using the "group law" (trapdoor function), but it is very difficult to work backward to "divide" a point by a number. In other words, assuming that it is intractable to factor a large integer composed of two or more large prime factors, the public key systems are secure. The greatest common divisor of two numbers can be found by using the Euclidean algorithm.

The RSA algorithm can be slow when generating key pairs with large primes. Besides this, when encrypting large data in the same computer, the RSA algorithm can also be very slow. For further explanation, the main computational cost of the RSA algorithm is the modular exponentiation during the key generation, encryption and decryption process. The reason is that this algorithm needs a third trusted party to identify the public keys. During the data transmission, it can be exposed to the middlemen who is able to temper with the public key system and the algorithm can be compromised. Thus, a secure implement is difficult due to the slow speed of signing and decryption. In addition, the RSA algorithm has weaknesses against certain attacks, such as Brute force as the capacity of supercomputer advances rapidly.

3. PERFORMANCE ANALYSIS

In this section, we mainly discuss the complexity and performance of the proposed authorized encrypted search scheme for multi-authority medical databases. We implement the proposed scheme through Java in eclipse platform. The experiment is conducted on a laptop with the Windows 11 operation system, Intel Core i5. To realize non-interactive token generation, we deploy an RSA protocol to the system setup. Our RSA implementation uses the insider Java cryptographic library.

Overhead Analysis

The overhead in our scheme is mainly concentrated on three parts, i.e., a multi-authority attribute-based encryption policy for fine-grained access control; a hash keyword to prime function for mapping keyword to prime; and an RSA function for achieving non-interactive functions. we know that the cost of the hash keyword to prime function is a one-time operation, and we can build a table to record all the possible mapping for later use.

4. CONSTRUCTIONS

There are four main algorithms of a CP-ABE scheme, which respectively are setup, key generation, encryption and decryption. In the first setup algorithm, two prime numbers are picked based on the concept of RSA algorithm, and new security parameters are produced step by step from these two primes.

Notation	Description
	λ the security parameter
	$id \in \{0,1\}^{\lambda}$ the document identifier
w	set of authorized keywords for the client
DB= (id_i,W_{idi})	database form of document-keyword pairs
EDB= (u,e)	encrypted database
DB(w)	set of documents that contain keyword w
I	set of document identifier

A*i*	access policy for authority *i*	
$S\ \sigma$	client's attributes(client's global identity)	
L	, '0' denotes add; and '1' denotes delete	
	the leakage function	
	$New(w)$	result of new query
	$Old(w)$	result of previous query
	$\sigma \in \{0,1\}$	

After getting all the security parameters, a master public-key as well as a master secret key are created by utilising the security parameters produced in the setup stage. In the second algorithm, it takes a set of attributes and the master private key as inputs and then it outputs a secret key associated with a set of attributes defined by each decoder by using a hash function for each attribute. In the third algorithm, a Lagrange polynomial is selected with an up-to-down manner for each node in an access structure.

Notations

Here, we present an authorized encrypted search scheme for multi-authority medical databases. Note that, all the keywords have already been mapped to the available prime integers by Algorithm 1 in the scheme. Let F: $\{0,1\}^{\lambda} \times \{0,1\}^{*} \rightarrow \{0,1\}^{\lambda}$ be a key-based pseudo random function and $H_1: \{0,1\}^{*} \rightarrow \{0,1\}^{\lambda}$,$H_2: \{0,1\}^{*} \rightarrow \{0,1\}^{2\lambda+1}$ be two cryptographic hash functions, and **P** be a symmetric pseudo-random permutation. Then our multi-authority encrypted databases can be described as follows:

Setup: In this stage, each authority executes independently to initialize the system and generates an encrypted medical database for their existing database. Without loss of generality, we take the *i*-th authority as an example. It takes the security parameter λ and a local database DB_i as inputs, and then outputs its public key, master key and an encrypted database EDB_i. Specifically, the *i*-th authority first chooses two big prime integers p_i,q_i and sets $N_i = p_iq_i$, then it selects a string k_i from $\{0,1\}^{\lambda}$ randomly. Let g be an element in group G. At the same time, it also needs to perform an MA-ABE setup protocol to obtain a pair of keys (mpk_i,msk_i), where (mpk_i,msk_i) ← **MA-ABE.Setup**(1^{λ}). Then the public key and the master key of the *i*-th authority are $PK_i = (N_i,F,H_1,H_2,\mathbf{P},mpk_i)$ and $MK_i = (p_i,q_i,k_i,g,msk_i)$, respectively. Finally, the *i*-th authority takes PK_i,MK_i, DB_i and a set of access policies of interested authorities $\bigcup_{i=1}^{l}\{\mathbb{A}_i\}$ as inputs, and then outputs an encrypted database EDB_i and a map $\mathbf{T}_i$ which maps keyword to the encrypted partial token by Algorithm 2.

Algorithm 2 Generate encrypted database

Input: *MK*, *PK*, DB, ; $S \subseteq_{i=1}^{l} \{A_i\}$ $S \subseteq_{i=1}^{l} \{A_i\}$

// here denotes the set of access policies of the *l* authorities involved in encryption

Output: EDB, **T**

1: EDB ← {},**T** ← empty map
2: **for** $w \in$ W **do**
$c \leftarrow 0, st_0 \leftarrow\!\!- \{\$0,1\}^{\lambda}$
3:
4: $stag_w \leftarrow F(k, g^{1/w} \bmod n)$
5: **for** $id \in$ DB(w) **do**
6: $c \leftarrow c + 1$
$t_c \leftarrow\!\!- \{\$0,1\}^{\lambda}$
7:
8: $st_c \leftarrow \mathbf{P}(t_c, st_{c-1})$
9: $u \leftarrow H_1(stag_w || st_c)$
10: $e \leftarrow (id||0||t_c) \oplus H_2(stag_w||st_c)$
11: EDB[u] = (e)
12: **end for**
13: **T**[w] ← **MA-ABE.Enc**$(\bigcup_{i=1}^{l}\{mpk_i\}, st_c||c, \bigcup_{i=1}^{l}\{\mathbb{A}_i\})$
14: **end for**
15: **return** EDB, **T**

ClientKGen: Similar to the Setup stage, here we also take the *i*-th authority as an example to introduce how to generate a private key for the client with attributes S and a set of authorized keywords $\mathbf{w} = (w_1,w_2,...,w_n)$.

First it computes $sk\mathbf{w} = (sk\mathbf{w},1,sk\mathbf{w},2) \leftarrow (k_i, g^{1/Q_{nj=1}\ w_j} \bmod N_i)$, and then executes MA-ABE schemes to obtain the attributes key $sk_S \leftarrow$ **MA-ABE.KeyGen**(msk_i,S). Finally, it sends the private key $sk = (k_i, g^{1/Q_{nj=1}\ w_j} \bmod N_i, sk_S)$ to the client together with the authorized keyword set **w** in a secure channel.

Encrypted Data Search: Whenever the client described above wants to search the documents with keyword w over the database EDB_i, she first scans the mapping table $\mathbf{T}_i$ to fetch the hidden value eck := **MA-ABE.Enc** $(\bigcup_{i=1}^{l}\{mpk_i\}, st_c||c, \bigcup_{i=1}^{l}\{\mathbb{A}_i\}) \leftarrow \mathbf{T}_i[w]$ and then decrypt-s it to obtain $st_c||c \leftarrow$ **MA-ABE.Dec**(sk_S,eck) with her attributes secret key sk_S. If the client can get $st_c||c$, she will have search permissions; otherwise, she will not. After that, the client computes $stag_w \leftarrow F(sk_{\mathbf{w},1}, sk_{\mathbf{w},2}^{\prod_{w_i \in \mathbf{w}/w}\ w_i} \bmod N_i)$ for the

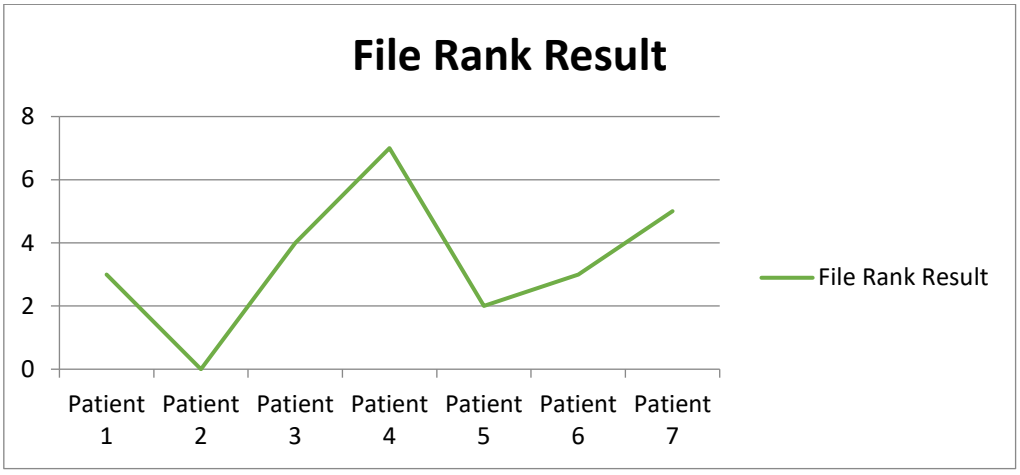
expected keyword w. Finally, she sends the total search token $ST_w \leftarrow (stag_w, st_c \| c)$ for keyword w to the server. Once the server receives the client's search token, she takes encrypted database EDB, and search token $ST_w \leftarrow (stag_w, st_c \| c)$ as inputs, and then initializes two empty sets I and D. Finally, the server performs Algorithm 3 to get the set of document identifiers I, and returns it to the client.

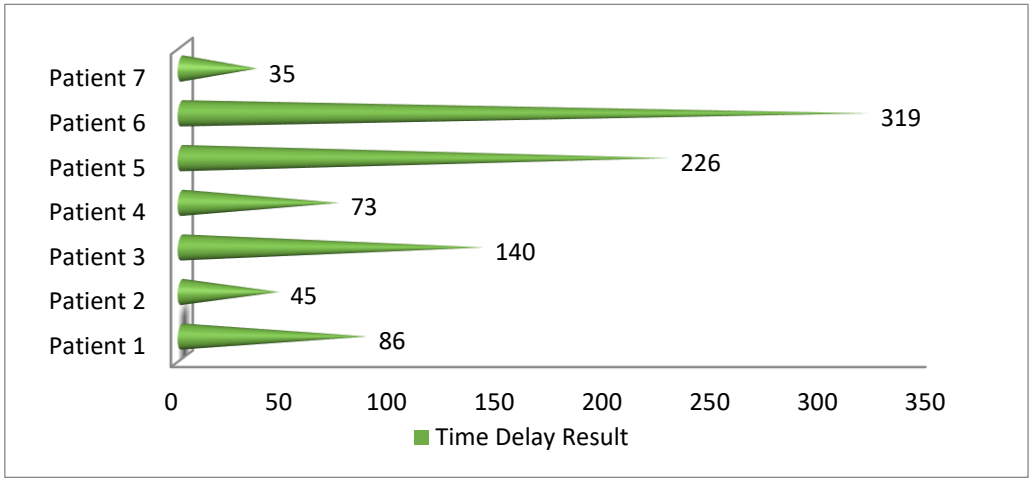
Algorithm 3 Encrypted data search
Input: Search token:ST_w, Encrypted Database: EDB
Output: I
1: I ← {},D ← {}
2: $(stag_w, st_c \| c) \leftarrow ST_w$ 3: **for** $i = c$ to 1 **do**
4: $u \leftarrow H_1(stag_w \| st_i)$
5: $e \leftarrow EDB.find(u)$
6: $id \| \sigma \| t_i \leftarrow e \oplus H_2(stag_w \| st_i)$
7: **if** $\sigma = 1$ **then**
8: D ← D ∪ {id}
9: **else**
10: I ← I ∪ {id}
11: **end if**
12: $st_{i-1} \leftarrow \mathbf{P}^{-1}(t_i, st_i)$ // $\mathbf{P}^{-1}$ denotes the inversion function of **P**
13: **end for**
14: I ← I − D
15: **return** I

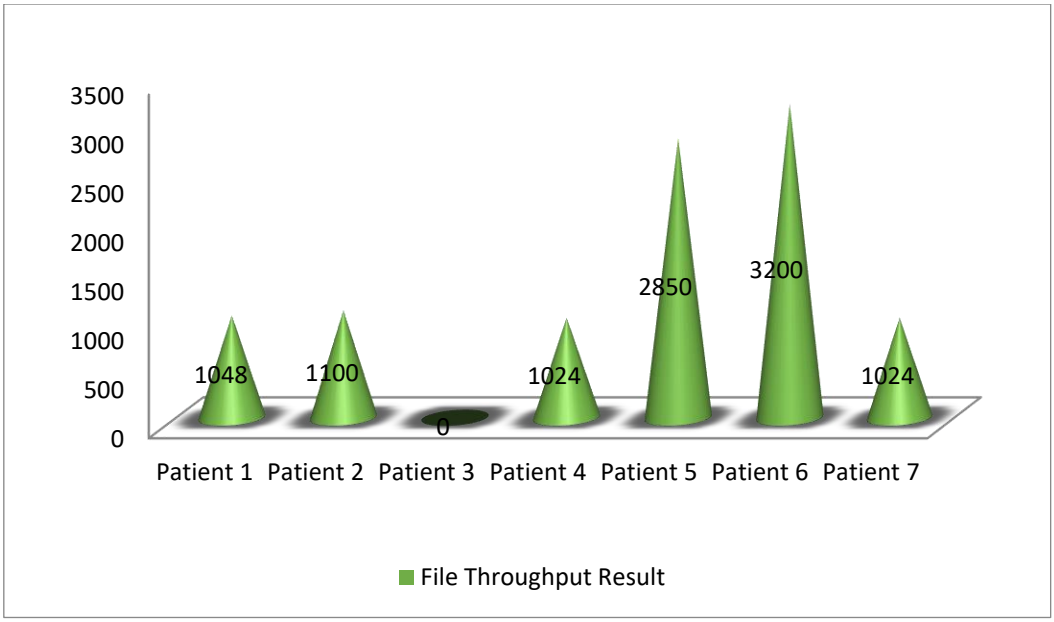
Update Encrypted Database. To allow for addition and deletion of the documents, we consider dynamic encrypted search databases. When the i-th authority wants to add a new pair (id,w) into the database, it takes its own master key MK_i, public key PK_i, encrypted database EDB_i and the document-keyword pair (id,w) as inputs, and then executes Algorithm 4 to encrypt the new pair and inserts the encrypted copy into the encrypted database EDB_i. During this process, the authority needs to renew the corresponding mapping table $\mathbf{T}_i$ as well.

Mark.Note that, the mapping table **T** in our system can be published or stored at another trusted server which will not collude with the server mentioned before. In addition, to improve this process, we can also generate another token for the keyword to hide its information.

5. ANALYSIS







6. AKNOWLEDGMENT

I would like to express my deep and sincere gratitude to my college DR. M.G.R EDUCATIONAL INSTITUTE, CHENNAI and primary research supervisor Associate Dr. T. V Ananthan and Dr. Dinesh Kumar for giving me the opportunity to do this research and providing invaluable guidance throughout this research and making the CP-ABE projects possible. Her extensive industry and academic experiences have been extremely valuable in contributing to the successful completion of my projects and thesis. It was a great privilege and honour to study under her guidance. I am extremely grateful for what she has offered me. I would also like to thank for her kindness and patience during the whole process. I would also like to express my appreciation to my friends

7. REFERENCES

[1] S. Jarecki, C. S. Jutla, H. Krawczyk, M. Rosu, and M. Steiner, "Outsourced symmetric private information retrieval," in Proc . of 2013 ACM SIGSAC Conf. on Comput. and Commun. Secur., 2013, pp. 875–888.
[2] P. Xu, S. Liang, W. Wang, W. Susilo, Q. Wu, and H. Jin, "Dynamic searchable symmetric Privacy, 2017, pp. 207–226.
[3] R. Bost, "Poϕoς: Forward secure searchable encryption," in Proc. of the 2016 ACM SIGSAC Conf. on Comput. and Commun. Secur., 2016, pp. 1143–1154.
[4] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. of Int. Conf. on the Theory and Appl. of Cryptographic Tech., 2004, pp. 506–522. [21] L. Fang, W. Susilo, C. Ge, and J. Wang, "Public key encryption with keyword search secure against keyword guessing attacks without random oracle," Inf. Sci., vol. 238, pp. 221–241, 2013.
[5] L. Zhou, V. Varadharajan, and M. Hitchens, "Achieving secure role-based access control on encrypted data in cloud storage," IEEE Trans. Inf. Forensics Secur., vol. 8, no. 12, pp. 1947–1960, 2013.
[6] C. Zuo, J. Shao, Z. Liu, Y. Ling, and G. Wei, "Hidden-token searchable public-key encryption," in Proc. of 2017 IEEE Trustcom/BigDataSE/ICESS, 2017, pp. 248–254.
[7] Y. Guo, X. Yuan, X. Wang, C. Wang, B. Li, and X. Jia, "Enabling encrypted rich queries in distributed key-value stores," IEEE Trans. on Parallel and Distrib. Syst., 2018.
[8] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in Proc. of 2007 IEEE Symp. on Secur. and Privacy, 2007, pp. 321–334.
[9] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. of the 13th ACM Conf. on Comput. and Comm. Secur., 2006, pp. 89–98.

