Applying Multi Property Tree for Multi Keyword Rank Searching and Dynamic Update in Cloud

Pawan Kumar Tanwar, Ajay Khunteta, Vishal Goar

Abstract**: *Various types of data structures are used for keyword searching like binary tee, KBB tree, inverted tree, inverted index, Multi Property Tree (MPT). These data structures are used for searching keywords in cloud space after getting instructions from data user. On the basis of MPT data structure the authors have introduced a search scheme called MPTsearch algorithm. Experiments show that the proposed scheme performs better than linear search. It also achieves lower time consumption and computing overhead for queries and trapdoor formation. Moreover this scheme not only fulfills the searching part but also plays vital role in dynamic update (insertion and deletion) of data provided by the data owner.***

Keywords : Multi Property Tree, MPTsearch algorithm, dynamic update.

I. INTRODUCTION

Cloud computing plays a very significant role in our life due to its storage place and anytime availability of data and computing resources at very low cost. Due to these advantages many enterprises and human being keep their important data at cloud space instead of buying local computing resources for managing the data. Data security and privacy is also very important issue. For the protection of security of data and privacy of user, data encryption is mandatory before uploading the data upon cloud. The encrypted data is stored in the form of cipher text.

The basic form of data usability is the keyword searching from large amount of data. There are so many keyword searching schemes are available. One practical approach for keyword searching is SE (searchable encryption)[16][17].SE schemes support keyword searching upon encrypted cloud data and simultaneously saves bandwidth of network and computing resources.

Till today so many SE schemes have been developed by researchers for searching keywords. Some of them are single word, multi keyword, similarity, rank search and many more. In all of these schemes multi keyword rank search bases on MPT(Multi Property Tree) is far better than other schemes for large quantity records like social site data, electronic mail, digital health data(DHD).

For example due to increasing number of patient in hospitals they are bound to keep the records of each and every patient's data at cloud server for so many purposes. Here every record is composed of so many properties or attributes of a patient like age, sex, disease, place etc. For extracting the output of a query sometimes we need to club more than one attribute of a patient which is known as conjunctive keyword search.

For the purpose of making some statistical report the authorized user can get all the records of the patient with "(55<=age<=75) and disease = "asthma". For enhancing the searching procedure, multi property conjunctive keyword searching method with subset and range query have been proposed.

II. RELATED WORK

With the help of SE schemes the data owner able to upload their encrypted data at cloud server and similarly the data user can search the keywords from the cipher text. Numbers of searchable encryption schemes were suggested for various threat models exists to get different searching functions. Symmetric key or public key cryptography provides the basis for SE schemes.

As per the searching function, searchable encryption schemes deals in mainly two types single and multi keyword searching.

1. Single Keyword Searching

On the basis of symmetric key cryptography song et al. [1] has suggested single keyword SE scheme. The timing complexity of the scheme was O(m), here m is data group size. Optimal timing complexity has been achieved for searching in two schemes proposed by [3].

On the basis of public key cryptography SE scheme has been provided by [4]. In the scheme of [4] the authorized user with private key can do searching. All these schemes are applied for single keyword searching and not able to searching for a group of keyword.

2. Multi Keyword Searching

Multi keyword rank searching and multi keyword boolean searching are two basic type of multi keyword searching. Here a group of keywords are searched and top k files related to keywords are returned. In conjunctive search those files are returned which are containing all the searched keywords. Golle et al. [13] very first provided the conjunctive search based on symmetric key cryptography.

Revised Manuscript Received on September 25, 2019

Pawan Kumar Tanwar, Research Scholar, Dept. of Comp. Science, Poornima Univ. Jaipur & Asst.Prof., Govt. Engg. College,Bikaner,India
Dr. Ajay Khunteta, Professor,Dept. of Comp. Science, Poornima Univ. Jaipur, India
Dr. Vishal Goar, Asst. Professor, Dept. of Comp. Appl., Govt. Engg.College, Bikaner, India.

Ballard et al. [7] proposed two conjunctive searching schemes to reduce the computing cost and trapdoor size.

On the basis of public key cryptography Park and Lee et al. [8] proposed the conjunctive searching schemes. Liu et al. [10] provided the conjunctive keyword searching scheme for large number of files. Boneh and Waters [6] constructed a public key searchable scheme which supports equality conjunction by extending the function of conjunctive keyword. Shen et al. [5] provided a scheme for multi dimensional range query for cipher text. Based on bilinear pairing these two schemes require high cost for computing.

Zhang et al. [18] formed a multi keyword conjunctive system for searching that supports range, equality and subset conjunction. Disjunctive searching schemes provides all files which includes a bunch of keywords queried. In near past some encryption schemes were proposed for supporting disjunctive searching. Katz et al. [12] suggested a scheme with predicate encryption that is supporting disjunction, polynomial equation and inner products. The k (top) most relevant documents are outputted in ranked search. Cao et al. [14] suggested the multi keyword rank searching scheme for privacy preservation. Here dictionary size includes the query and files.

As the importance of distinct keyword was not considered hence the scheme wasn't perfect. Additionally the complexity of searching was linear with size of data. A searchable index based on tree structure was constructed by Sun et al. [19][20] with adopting cosine measure along with tf x idf.

The author in [11] provided a multi keyword rank searching scheme with additive ordering preservation function to get the highly relevant searching output. Update is performed by the owner of data after uploading the documents at cloud space. Hence support for document updation is expected from SE schemes.

Till date so many dynamic SE methods have been proposed. To execute dynamic update scheme for encrypted file group efficiently the inverted index is formed by Kamara et al. [7] but the execution of the scheme is very tedious. Papamanthou and Kamara [4] proposed a KRB tree based method for the dynamic updation of documents after some time. These methods deals in only single keyword boolean searching. The author in [15] provided a multi keyword rank searching method for flexible file updation. Because of adopting the tree based indexing method by [15], the sub linear searching time is achieved.

In this paper the authors were designed a MKRSCD (multi keyword ranked search scheme upon cloud data) scheme, particularly enforcing upon query and index. At next step they have provided another enhanced scheme MKRSCD-CM (Multi keyword ranked search scheme for cloud data in known cipher text model). To get the accurate output, thorough evaluation of security has been done and the experiments done on the real set of data. The evaluation shows that the scheme is designed according to meet the design objectives [21].

Jonker et al. [2] described a scheme for the wildcard searching upon encrypted cloud data for making searching queries much flexible. The author developed the scheme by applying pseudorandom functions and Bloom filters.

The author proposed a framework for forming and analyzing public key system which supports comparison query upon enciphered data and more queries like subset query [6].

The authors studied the problem of a PECK (public key encryption with conjunctive keyword) searching. This method enabling an user to upload the files a non trusted cloud server and having the ability of selection searching the information without leak of data [9].

The authors explained few schemes of cryptography like B-tree and CKRSA for enhancing the security level that leading to trust [23].

Two schemes called BDMRS-CM (Basic dynamic multi-keyword ranked search scheme in the known ciphertext model) with the use of secure kNN algo and the EDMRS-BM (enhanced dynamic multi-keyword ranked Search scheme in the known background model) were designed. The authors have also done evaluation and analysis of both the schemes. The output of practicals demonstrates that the designed scheme would enable the multi keyword rank searching in a better way [22].

The related overwork provides a brief overview of the existing searching schemes available.

III. Problem Formulation

The formulation of the problem is described below:

1. System model

System model includes the owner, cloud and user. We are taking Digital Health Data (DHD) to understand the things. The hospital could be considered as owner of data here. The owner outsource the encrypted documents F = {f1……….fn} with index tree T constructed by queries Q={q1…………qn} to server.

For update the owner is partially responsible. The data owners makes the modifications in his machine then outsource the data at server.

The user performs the search in enciphered files generally. The authenticate data user could construct the trapdoor from query keywords as per the searching mechanism. After receiving the matched encrypted files, user can decipher the files by using shared key (secret).The server keeps the enciphered files and index tree. As per the particular query protocol search will be done by the cloud server after getting the search trapdoor.

The authors basically emphasizes upon multiple property conjunctive search in cipher text. Particularly the emphasis is upon construction, encryption and search upon index tree. The cloud server is treated here as semi honest according to [13], [14], [18].

Attack model

As per the cloud server data the following attack models are taken into consideration.

(i)Cipher text attack model

Here the server keeps the knowledge about the set of encrypted files, encrypted index tree and trapdoor introduced by data user.

(ii) Plaintext attack model

Here the server keeps the information about set of records in the properties vector of DHD and enciphered values of these records excluding the knowledge in cipher text attack model.

(iii) Known background model

Here server keeps some background information of set of data except the data in plaintext model. Peculiarly attacker might have few data like vector, keywords, etc.

2. Design objectives

For the execution of complex, advanced and secured multi property conjunctive search upon encrypted DHD, the author's schemes are aiming to get the following objectives:

a) Privacy

To get the privacy objective the server should not know metadata. The sub objectives of privacy are- (1) Index and Query Privacy – To maintain this privacy we have to stop the server from getting information of plaintext relevant to encrypted index and query. (2) Trapdoor Unlinking – To get this objective the server could not be able to get the knowledge that if 2 trapdoors were formed of repeated searching. (3) Keyword Privacy – The server could not determine keyword and properties from existing information of database.

b) Efficiency

The author's scheme is aimed to achieve far better search efficiency than linear search by retrieving the information from a MPT based index and an algorithm for search upon the tree (index).

c) Dynamic updation

The authors aimed to achieve dynamic updation (deletion and insertion) of files.

d) Real world searching

The objective of real world searching can be achieved by applying data matching and association rules upon encrypted DHD.

3. Preliminaries

1. Symbol

P – It is the property collection, described a group of m properties, P ={P1,………..Pm)

R – Plaintext record set, defined as group of n record, R = {r1,……….,rn}. Every record in F could be seen as an ordered j- record {j1…….jm} of values where jm relative to property Pi.

F- The plaintext documents group, described as a group of n files, F = {f1,…………..,fn}, where fn relevant to rn.

E – It is encrypted document set, kept in server, described as a group of enciphered files, E = {e1,………..,en}, where ei denoting the enciphered documents of fi.

S – The searchable encrypted index constructed as per the record group R.

2. Multi Property Tree (MPT)

For solution of the conjunctive keyword query problem, we use the data structure called MPT. A multi dimensional MPT formed on the basis of n attributes of records. A tree T is constructed with n levels. Basically the MPT is constructed as follows:

1. The MPT level is set according to the cardinality of property collection P with Pj relevant to level j. particularly the root level is 0 and the n properties of record relevant to next n level of MPT. The root vertex has assigned nothing. The other nodes/vertices of MPT are weighted with the relative property values.

2. According to the attribute values the records are being sorted in Pi (I =1, ……… m). Arrange the succeeding sort values in a single node column wise.

3. Set all values of keywords in the nodes of the tree structure. All the children vertices at next level are formed from the previous level.

4. Records are arranged as per ID values.

Id	Age	gender	disease	State
1	72	m	asthma	tn
2	62	f	tb	Jk
3	52	m	tb	Jk
4	72	m	dengue	tn
5	62	f	polio	tn
6	72	f	cancer	Hr
7	72	f	cancer	Hr
8	72	m	dengue	tn
9	62	m	cancer	Hr
10	72	f	cancer	Jk

Id	Age	gender	disease	State
3	52	m	tb	Jk
9	62	m	cancer	Hr
1	72	m	asthma	Jk
4	72	m	dengue	tn
8	72	m	Dengue	tn
2	62	F	tb	Jk
5	62	F	polio	tn
6	72	F	cancer	Hr

7	72	F	Cancer	Hr
10	72	F	Cancer	Jk

Table-1-Real & Sorted DHD

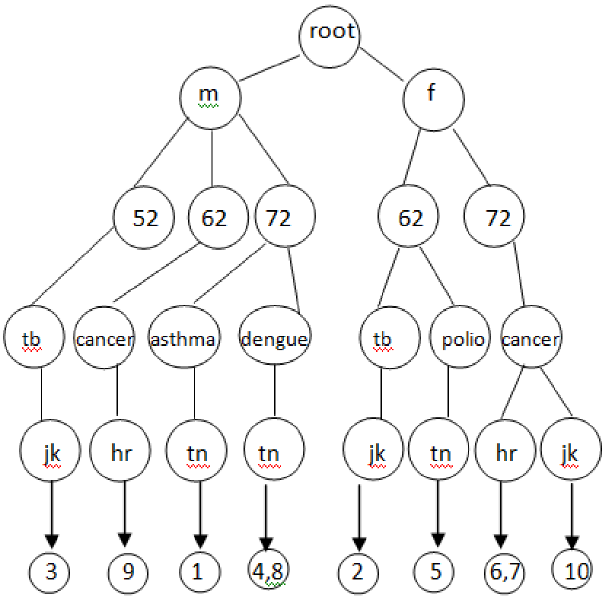

Figure-1 MPT Structure

The figure one shows the structure of the multi property tree (MPT).Here the root node has assigned no value but in the next level the nodes are assigned the values as per the attributes taken in the database of table-1 shown above.

IV. MPT SEARCH ALGORITHM

The authors proposed the search algorithm whose searching method is a recursion process upon MPT index known as MPTsearch algorithm. The proposed algorithm initiates searching in the MPT index at each stage from root, retrieving appropriate node at every level. The authors used the ordering of vertex keywords to represent the identification of group at level j preceded from a single vertex at level (j-1). For example in Figure-1 (j1,j2) = (m,72) is the identification of the group in the 3rd level, that includes the vertex keyword “asthma” & “dengue”.

STEPS OF ALGORITHM

```
START
Search level by level from root in mpt (multi prop. Tree) index (left to right) collect qualified node at each level match the keyword & query
If output=1
   the keyword matched with the query
   if
The keyword is at bottom level where the pointer points at id array  search method inserts record id from id array to output list h
else if
Keyword residing at middle level        search method goes to next level and  start searching left to right
else if output  != 1
The search method choses the right vertex of the keyword set
    END
```

V. DYNAMIC UPDATION METHOD

The authors take care of dynamic insertion and deletion of documents as in schemes [15] [17]. After insert and delete of documents the index must be updated. By updating the relevant nodes in the MPT index, the update method is completed because the index of proposed scheme is formed as MPT formation. The update of index is based on identification of documents. To update the index following procedures are required-

(i)Gendataupdate (VK, Ts, j, updkind)

updkind = type of update insert/delete,
j = identification of row/record
Ts = Subtree formed of tree nodes/vertices
VK = Secret Key to generate subtree Ts

Suppose the authors want to delete the row r1 in table-1, Ts refers to the set {asthma, tn, array ID :{ where ID=1}}

Two ways considered for operation of delete. First way is that if one or more rows in the group of records having the similar m property value as the deleted record rj then delete rj‘s ID from the array master ID. In the second way it is necessary to check every level keyword of the erased record rj. The authors remark the nodes by the node keyword for the convenience. Suppose the authors have to delete the record rj in figure-1. In the group of records there might be number of rows with keyword “m” at first level in the tree so the node “m” will be not edited. Similarly in the database number of rows might have keyword pair {m, 72}, so the node “72” will be not edited. However row t1 is the only row which satisfies the primary 3 level keywords are {m, 72, asthma} so the node “asthma” required to be erased. Similarly it is required to erase the node “tn” and the array ID with ID=1 is saved. Finally, ei is set as NULL which is enciphered file of fi relevant to record rj.

Similar to the deletion the insertion can also be done in 2 ways. First method is that if one or more rows in the database with the similar m words as the row of insert rj’s ID in the master list of ID. In the second method it is needed to check each level keyword of inserted record rj. Suppose the author want to insert row 11 as r11 if fig-1 having words {f, 62, cancer, tn}. In fig-1 two rows r2 and r5 having the keywords {f, 62}, so the node “f” and “62” will not be edited. In figure-1 there is no row whose primary level keywords are {f, 62, cancer}, so it is needed to append the vertex “cancer” as a child vertex of “62”. In the same way it is necessary to append “tn” as a child vertex of “cancer” pointing to a newly appended ID array with ID=11.

The owner enciphers the vectors of the newly appended keywords with secret key VK to form enciphered sub tree Ts. Finally the row fi is enciphered to ei by the enciphered way of files. As the output of providing the random digit in the procedure of encryption is indeterminate. It means that equal keyword is enciphered in distinct cipher text.

(ii) updt(I, E, updkind, Ts’, ei)

I = Index
E = Encrypted Document Collection/Group
Updkind = type of update insert/delete
Ts’= Encrypted form of subtree
ei = encrypted document

Update operation is performed by the cloud server after getting data for update from the owner of data. The cloud server deletes Ts’(encrypted Ts) from the index structure for generating an index I’ and deletes ei from the encrypted group of documents. If the update type is insert then the server appends the enciphered file ei into E for generating a new cipher text bunch E’ along with it inserts new sub tree Ts’ to generate a new structure of index I’. When MPT index is constructed it is necessary to apply identification for every node so the server might be able to get the information that a new sub tree is inserted or deleted.

VI. CONCLUSION

The conclusion is that the proposed scheme is far better than the other schemes discussed. Experiments demonstrate that our scheme performs much better than linear search. It also achieves lower time consumption and computing overhead for queries and trapdoor formation. The proposed scheme executes multi keyword rank search based on multi property tree along with the dynamic updation very effectively.

REFERENCES

1. Perrig, D. Wagner and D. X. Song ‘‘Practical techniques for searches on encrypted data,’’ In the Proc. of ieee sympo. for Sec. and Priva., Berkeley, USA, May 2000, pp. 44-55.
2. W. Jonker, Hartel P. and R. Brinkman, “Conjunctive wildcard search over encrypted data,” In the proceedings of workshop for secure data management, Seattle, USA, 2011, pp. 114-27.
3. R. Ostrovsky, S. Kamara, J. Garay and R. Curtmola, “Searchable symmetric ncryption: Improved definitions and efficient constructions,” Jou. of Comp. Sec., volume 19, number 5, pp. 895-934, Jan. 2011.
4. G. Persiano, R. Ostrovsky, G. Di Crescenzo and D. Boneh, “Public key encryption with keyword search”, in Proceedings of euro crypt, Switzerland, 2004, pp. 506-22.
5. Waters, Shi E. and E. Shen, “Predicate privacy in encryption systems”, in Th. of Crypt., San Francisco, USA, Springer, 2009, pp. 457-73.
6. B. Waters and D. Boneh, “Conjunctive subset and range queries on encrypted data,” In the Proceedings of Theory crypt. Conference, The Netherlands, 2007, pp. 535-54.
7. F. Monrose, Kamara S. and L. Ballard, “Achieving efficient conjunctive keyword searches over encrypted data,” in the proceedings of International Conference for Info. and Comm. Sec., China, 2005, pp. 414-26.
8. P. J. Lee, D. J. Park and K. Kim, “Public key encryption with conjunctive field keyword search,” in Proceedings of Int. workshop of Info. Sec. Application, South Korea, 2004, pp. 73-86.
9. P. J. Lee and Y. H. Hwang, “Public key encryption with conjunctive keyword search and its extension to a multi- user system,” In Proc. of Int. Conf. of Pairing Based Crypt., Tokyo, Japan, 2007, pp. 2-22.
10. J. Chen, L. Zhu and C. Liu, “Efficient searchable symmetric encryption for storing multiple source data on cloud,” in the proceedings of IEEE trust com / Big Data SE/ISPA, FINLAND, August 2015, pp. 451-58.
11. S. Zhou, J. Wu, Lin Y., Xiao S. and W. Zhang, “Privacy preserving ranked multi keyword search for multiple data owners in cloud computing,” ieee Transaction for Computing, volume 65, number. 5, pp. 1566-77, May 2016.
12. Waters, Sahai A. & J. Katz, “Predicate encryption supporting disjunctions, polynomial equations, and inner products,” In proceedings of International Conference, Theory Application Cryptography Tech., Turkey, 2008, pp. 146-62.
13. B. Waters, Staddon J. & P. Golle, “Secure conjunctive keyword search over encrypted data,” In the proceedings of International Conference, Application of Crypt. and Network Sec., Heidelberg, Germany, 2004, pp. 31-45.
14. W. Lou, K. Ren, M. Li, C. Wang and N. Cao, “Privacy-preserving multi-keyword ranked search over encrypted cloud data” in Proceedings of ieee infocom, shanghai, China, Apr. 2011, pp. 829-37.
15. Q. Wang, X. Sun, X. Wang and Z. Xia, “A secure and dynamic multi keyword ranked search scheme over encrypted cloud data,” Ieee Transaction for Para. Distr. Sys., volume 27, Number 2, pp. 340-52, Jan. 2016.
16. Roeder, Papamanthou C. & S. Kamara, “Dynamic searchable symmetric encrypt.”, in proceedings of ACM Conf. for Comp. and Communication System, Raleigh,USA, 2012, pp. 965-76.
17. Papamanthou and Kamara S., “Parallel and dynamic searchable symmetric encryption,” in proceedings of Int. Conf. for Fin. Crypt. & Data Sec., 2013, pp. 258-74.
18. Liu X. F., Quan H. Y., Y. Q. Zhang and L. L. Zhang ‘‘Efficient conjunctive keyword search over encrypted medical records,” (Chinese), J. Software, vol. 27, no. 6, pp. 1577-91, June 2016
19. W. Sun et al., “Privacy preserving multi-keyword text search in the cloud supporting similarity-based ranking,” in proceed. of ASIACCS, Hangzhou, China, 2013, pp. 71–82
20. Y. T. Hou, W. Lou, H. Li and W. Sun, “Privacy preserving keyword search over encrypted data in cloud computing,” in secure cloud computing, USA: Springer-Verlag, 2014, pp. 189–212
21. Tanwar Pawan Kumar, Goar Vishal, Khunteta Ajay, “Design of new multi keyword ranked search scheme and validation for cloud computing,” in proc. of AICTC - 16, Aug. 12 &13, 2016, Bikaner, India
22. Tanwar Pawan Kumar, Goar Vishal, Khunteta Ajay, “Performance evaluation of multi keyword ranked search schema
23. called BDMRS-CM & EDMRS-BM in cloud computing”, An Int. Journal of Engg. Sci., Issue July 2017, Vol. 24 pp 42-51
24. Tanwar Pawan Kumar, Goar Vishal, Khunteta Ajay, “Design and Analysis of Search Algorithm with B-tree and Commutative key RSA for dynamic Updation in Cloud Computing”, IJCAR (Int. Journal of Current Adv. Research) Vol 7, Issue 7(H), pp 14414-418, July 2018