International Journal of Science Engineering and Advance Technology

A Novel Approach For Using Multi-Keyword Ranked Search Scheme Over Encrypted in Secure And Dynamic Cloud Data

Syed Sadat Ali Alias Abdul Gani[1], Sayeed Yasin[2]
[1]Student, M.Tech (C.S.E), Nimra College of Engineering & Technology, A.P., India.
[2]Associate professor & Head, Dept. of Computer Science & Engineering, Nimra College of Engineering & Technology, A.P., India.

Abstract — cloud computing have revamped the view of modern information technology which is motivating the data owners to outsource their data to the public cloud server like Amazon, Microsoft Azure, Google Drive, etc. With the help of data outsourcing, the organizations can provide reliable data services to their users without any concerns for the data management overhead. . Normally, CSPs (Cloud Service Providers) take care of the data and its privacy, but there are some of the factors because of which the data privacy and user identity may be violated like an apostate employee, etc. In this way, information proprietors ought to scramble their particular touchy information before outsourcing it to people in general cloud server. Since the information is getting encoded before outsourcing which may influence the execution of some essential information getting to operations like seeking of a report, and so on. Searchable encryption is a cryptographic technique to give security. In writing numerous analysts have been chipping away at creating proficient searchable encryption plans. , we display a protected multi-watchword positioned look conspire over scrambled cloud information, which at the same time underpins dynamic overhaul operations like erasure and addition of records. In particular, the vector space display and the broadly utilized TF×IDF model are consolidated as a part of the file development and question era. We build an exceptional tree-based record structure and propose an "Eager Depth-first Search" calculation to give productive multi-catchphrase positioned look.

Keywords — Cloud Storage, multi Ranked-Search, Encrypted-Data Search, secure cloud.

I. INTRODUCTION

Distributed computing and capacity arrangements furnish clients and undertakings with different abilities to store and process their information in outsider information centers.[1] It depends on sharing of assets to accomplish cognizance and economy of scale, like an utility (like the power matrix) over a system. Distributed computing is a sort of Internet-based figuring that gives shared handling assets and information to PCs and different gadgets on request. It is a model for empowering omnipresent, on-request access to a mutual pool of configurable registering assets (e.g., systems, servers, stockpiling, applications and services),[2][3] which can be quickly provisioned and discharged with insignificant administration exertion. Security and protection concerns have been the real difficulties in distributed computing. The equipment and programming security systems like firewalls and so on have been utilized by cloud supplier. These arrangements are not adequate to shield information in cloud from unapproved clients as a result of low level of straightforwardness [4]. Since the cloud client and the cloud supplier are in the distinctive trusted area, the outsourced information might be presented to the vulnerabilities [5]. In this manner, before putting away the profitable information in cloud, the information should be scrambled [2]. Information encryption guarantees the information privacy and respectability. To save the information security we have to plan a searchable calculation that takes a shot at scrambled information [13]. Numerous scientists have been adding to looking on encoded information. The inquiry strategies might be single watchword pursuit or multi catchphrase look [11]. In immense database the inquiry may bring about numerous archives to be coordinated with watchwords. This makes trouble for a cloud client experience all reports and have most important records. Seek in view of positioning is another arrangement, wherein the reports are positioned in view of their pertinence to the watchwords [3]. Efficient searchable encryption methods help the cloud clients particularly in pay-as-you utilize show. The scientists consolidated the rank of records with numerous catchphrase pursuit to think of effective financially feasible searchable encryption systems. In searchable encryption related writing, calculation time and calculation overhead are the two most every now and again utilized parameters by the specialists as a part of the area for investigating the execution of their plans. Calculation time (additionally called "running time") is the time span required to play out a computational procedure for instance seeking a catchphrase, creating trapdoor and so forth.

II. PROPOSED SYSTEM

Encryption Module: By using CRSA, data in a file can be updated dynamically without affecting the overall performance of searching on B-tree. If the encrypted

indexed data is modified, re-indexing for the whole data is not needed. Similarly there is no need of re-encrypting the files in the database whenever the file is modified. This is a desirable feature as it reduces the computation time Commutative Encryption (CRSA): The RSA cryptosystem is one of the optimum public key cryptography approaches. However, its overall robustness gets limited due to one way encryption and majority of existing RSA schemes suffer from reorder issues. Therefore, in order to make this system least complicated and more efficient, an approach called Commutative RSA has been proposed. In this scheme, the order in which encryption has been done would not affect the decryption if it is done in the same order. Encryption is the standard method for making a communication private. With the many cryptographic approaches, our system follows the commutative RSA algorithm. Let us consider two prime numbers and initialized amongst all the group members. Let and represent the group members required to communicate over the documents. To compute the encryption keys and decryption key pairs of the commutative RSA algorithm the parameters and are computed using the following From the above equations it is clear that and for and . The encryption key pair of and are represented as (and is to be obtained. The is obtained by randomly selecting numbers such that it is a co-prime of or in other terms Where represents the greatest common divisor function between two variables and . The decryption key pair of and is represented by and and the parameter is computed based on the following equation Let represent the encrypted data . The encryption operation is defined as follows The commutative RSA decryption operation on the encrypted data is defined B- Tree: A B-tree is a data structure as shown in Figure 2. The tree contains index nodes and leaf nodes. All leaf nodes are at the same level (same depth). Each index nodes contain keywords and pointers. Each node except root node in a B-tree with order n must contain keys between n to 2n keys. Each node also contains (number of keys + 1) pointers to its child nodes. If the root node is an index node then it must have at least 2 children. The insertion, deletion, search operations takes only logarithmic time.

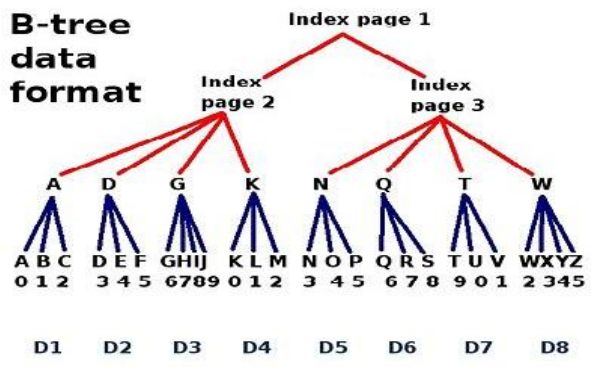

Figure 1: B tree data format.

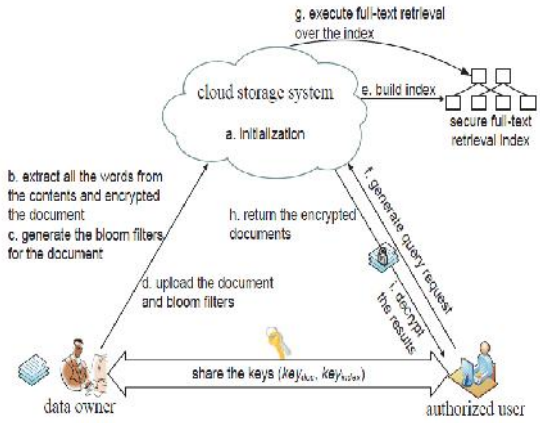

Figure 2: Architecture of searchable encryption scheme for multi-user databases.

III. LITERATURE SURVEY

Privacy-preserving Multi-keyword Text Search: Wenhai Sun [1] proposed this search that provides similarity based search result ranking, keyword privacy, Index and Query confidentiality and Query Unlink ability. The encrypted file is built by vector space model supporting consolidated and distinctive file search. The searchable index is built using Multidimensional B tree. Owner creates encrypted query vector $\bar{Q}$ for file keyword set. User gets the respective encrypted query vector of W from owner which is given to CS. Now CS searches index by Merkle–Damgård construction algorithm and compares cosine measure of file and query vector and returns top k encrypted files to user. Limitation: -The similarity rank score of the document vector fully depends on the type of the document.

A Secure and Dynamic Multi-keyword Ranked Search Scheme over Encrypted Cloud Data: This proposed method [2] suggest a secure tree-based search scheme over the encrypted cloud storage, which supports multi keyword ranked search along with dynamic operation on document collection available at server. The vector space model and term frequency (TF) × inverse document frequency (IDF) model are combinly used in the construction of index and generation of query to provide multi keyword ranked search output. To obtain high search efficiency results, author construct a tree-based index structure and proposed a Greedy Depth-first Search algorithm based on this index tree. Because of this special structure of tree-based index, the proposed search scheme can flexibly achieve sub linear search time and can effectively deal with the deletion and insertion of documents. The KNN algorithm is applied to encrypt the index and query vectors, and till then ensure accurate relevance score

calculation between encrypted index and query vectors.

Efficient and Secure Multi-Keyword Search on Encrypted Cloud Data: This proposed method has defined and solved the problem of effective but safe and sound rank keyword search over Encrypted cloud data [3]. Ranked search greatly enhances system usability by returning the matching files in a ranked order regarding to certain important criteria (e.g. keyword frequency) thus making one step closer towards sensible consumption of secure data hosting services in Cloud Computing. These papers has defined and solved the challenging problem of privacy preserving and efficient multi keyword ranked search over encrypted cloud data storage (MRSE), and establish a set of strict privacy requirements for such a protected cloud data utilization system to become a reality. The proposed ranking method proves to be efficient to go back extremely relevant documents corresponding to submitted search terms. The idea of proposed ranking method is used in our future system in order to enhance the security of information on Cloud Service Provider. Limitation: - Dynamic updating and deletion of the document from the cloud is not possible.

Privacy Preserving Multi-Keyword Ranked Search (MRSE): Ning [4] proposed this search for known cipher text model and background model over encrypted data providing low computation and communication overhead. The coordinate matching is chosen for multi-keyword search. They used inner product similarity to quantitatively evaluate similarity for ranking files. The drawback is that MRSE have small standard deviation σ which weakens keyword privacy. Limitation: - Multi-keyword ranked search (MRSE) for known cipher text model may produce two different trapdoor which vague the privacy leakage problem of trapdoor unlink ability which may weaken the keyword privacy. MRSE has small standard deviation σ which in turn weakens the keyword privacy. The integrity of the rank order is not checked in MRSE.

IV. RESULT ANALYSIS:

ALGORITHM-1 Btree_insert (root, Key, Object_value)
Input: root pageID of a B-tree, the key and the value of an object. //Inserts when Object_value doesn't exist in a B-tree
1. NODE = Disk_Read (root).
2. if NODE_x is full
   (a) y = Allocate_Page(), z = Allocate_Page().
   (b) Locate the middle object o stored in NODE_x.
   Move the objects to the left of object o into. NODE_y. Move the objects to the right of o into .
   NODE_z.
   If NODE_x is an index page,
   Then move the child pointers of NODE_x• accordingly.
   (c) NODE_ x: child [1] = NODE_y, NODE_x: child [2] = NODE_z.
   (d) Disk_Write (NODE_x); Disk_Write (NODE_y); Disk_Write (NODE_z).
3. end if
4. Insert_Not_Full (NODE_x; Key; Object_value).

The search performance is tested respectively by starting 1, 4, 8 and 16 threads. We compare the search efficiency of our scheme with that of Sun et al. [27]. In the implementation of Sun's code, we divide 4000 keywords into 50 levels. Thus, each level contains 80 keywords. According to [27], the higher level the query keywords reside, the higher the search efficiency is. In our experiment, we choose ten keywords from the 1st level (the highest level, the optimal case) for search efficiency comparison.However, when we continue to increase the threads, the search efficiency is not increased remarkably. Our search algorithm can be executed in parallel to improve the search efficiency. But all the started threads will share one result list RList in mutually exclusive manner. When we start too many threads, the threads will spend a lot of time for waiting to read and write the RList. An intuitive method to handle this problem is to construct multiple result lists. However, in our scheme, it will not help to improve the search efficiency a lot. It is because that we need to find k results for each result list and time complexity for retrieving each result list is O(θm log n/l). In this case, the multiple threads will not save much time, and selecting k results from the multiple result list will further increase the time consumption. In the Fig. 8, we show the time consumption when we start multiple threads with multiple result lists. The 5 10 15 20 25 30 35 40 0 50 100 150 200 # of documents in the collection(×102) Time of search(ms) Sun−1st level BDMRS EDMRS EDMRS with 4 threads EDMRS with 8 threads EDMRS with 16 threads (a) 20 40 60 80 100 0 50 100 150 200 Time of search(ms) # of retrieved documents Sun−1st level BDMRS EDMRS EDMRS with 4 threads EDMRS with 8 threads EDMRS with 16 threads (b) Fig. 7. The efficiency of a search with ten keywords of interest as input: (a) for the different sizes of document collection with the same dictionary, m = 4000, and (b) for different numbers of retrieved documents with the same document collection and dictionary, n = 1000, and m = 4000. 5 10 15 20 25 30 35 40 0 20 40 60 80 100 # of documents in the collection(102) Time of search(ms) EDMRS with 8 threads & 1 RList EDMRS with 16 threads & 1 RList

EDMRS with 8 threads & 2 RLists EDMRS with 16 threads & 4 RLists (a) 5 10 15 20 25 30 35 40 0 20 40 60 80 100 # of documents in the collection(102) Time of search(ms) EDMRS with 8 threads & 1 RList EDMRS with 16 threads & 1 RList EDMRS with 8 threads & 2 RLists EDMRS with 16 threads & 4 RLists

needed. Similarly there is no need of re-encrypting the files in the database whenever the file is modified. This is a desirable feature as it reduces the computation time. Data owner first generates secret and public key pair (EK, DK) using a standard public-key encryption scheme ie CRSA. Then owner makes the public key DK public and keeps the secret keys EK private. Documents {D | D1, D2,…, Dn} are encrypted using EK resulting in a ciphertexts {C | C1,C2,….Cn}. The generated C is stored in cloud database. The constructed index based on B tree is also encrypted using CRSA, i.e each derived keywords {W| w1,w2,….wn}from a document is indexed in a tree and encrypted using CRSA. This results in a set of encryptions {e| e1,e2,..en} where each ej (for) is defined as E_wj = CRSA_Enc (EK, wj), where E_wj denotes encrypted keyword.

V. CONCLUSION AND FUTURE WORK

The paper , We construct a special keyword balanced binary tree as the index, and propose a “Greedy Depth-first Search” algorithm to obtain better efficiency than linear search. In addition, the parallel search process can be carried out to further reduce the time cost. The security of the scheme is protected against two threat models by using the secure kNN algorithm. Experimental results demonstrate the efficiency of our proposed scheme.

Using CRSA, data in a file can be updated dynamically without affecting the overall performance of searching on B-tree. In our proposed system, if encrypted data is modified, re-encrypting for the whole data is not needed. This is a desirable feature as it reduces the computation time. The future work would concentrate on using Elliptic Curve Cryptography (ECC) encryption technique for better performance. Further, we intend to analyze the behavior of our proposed system(s) for multiuser environment.

VI. REFERENCES

[1] Qin Liuy, Guojun Wangyz, and Jie Wuz, ”Secure and privacy preserving Keyword searching for cloud storage services”, ELSEVIER Journal of Network and computer Applications, March 2011.

[2] Cong Wang et al., ”Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data”, IEEE Transactions on parallel and distributed systems, vol. 23, no. 8, August 2012.

[3] International Journal of Computer Applications (0975 – 8887) Volume 126 - No.14, September 2015.

[4] Wenhai Sun et al., "Privacy-Preserving Multikeyword Text Search in the Cloud Supporting Similarity-based Ranking", the 8th ACM Symposium on Information, Computer and Communications Security, Hangzhou, China, May 2013.

[5] Jiadi Yu, Peng Lu, Yanmin Zhu, Guangtao Xue, Member, IEEE Computer Society, and Minglu Li,”Toward Secure Multikeyword Top k Retrieval over Encrypted Cloud Data”, IEEE Journal of Theoretical and Applied Information Technology 10th August 2014.

[6] Ning Cao et al.,” Privacy-Preserving MultiKeyword Ranked Search over Encrypted Cloud Data”, IEEE Transactions on parallel and distributed systems, vol. 25, no. 1, Jan 2014.

[7] Wenhai Sun et al., "Protecting Your Right: Attributebased Keyword Search with Finegrained Owneren forced Search Authorization in the Cloud", IEEE INFOCOM 2014, Toronto, Canada, April 27-May 2, 2014.

[8] Secure Ranked Keyword Search over Encrypted Cloud Data, IEEE PAPER, 2010.

[10] Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, ‘Public key encryption with keyword search,’ in Proc. of EUROCRYPT, 2004.

[11] C.Wang et al., ‘Secure Ranked Keyword Search Over Encrypted Cloud Data,’ Proc. ICDCS ’10, 2010

[12] Wenjun Lu; Varna, A.L.; Min Wu, ‘Confidentiality-Preserving Image Search: A Comparative Study Between Homomorphic Encryption and Distance-Preserving Randomization,’ Access, IEEE, vol.2, no., pp.125,141, 2014

[13] W. K. Wong, D. W.Cheung, B. Kao, and N. Mamoulis, ‘Secure knn computation on encrypted databases,’ in Proc. of SIGMOD, 2009.

[14] K. Ren, C. Wang, and Q. Wang, ‘Security Challenges for the Public Cloud,’ IEEE Internet Computing, vol. 16, no. 1, pp. 69-73, 2012.

[15] Zhangjie Fu et al, ‘Multikeyword Ranked Search Supporting Synonym Query over Encrypted Data in Cloud Computing’, IEEE Conference, 2013.

Mr. Syed Sadat Ali Alias Abdul Gani, is a student of Nimra College of Engineering & Technology, Jupudi, Ibrahimpatnam, Vijayawada-521456. He is presently pursuing his M.Tech in CSE from JNTU, Kakinada. He has

obtained his Master of Computer Application (MCA) from Osmania University.

SAYEED YASIN received his M.TECH in Computer Science & Engg from JNTU Hyderabad. He is pursuing Ph.D., in Rayalaseema University, Kurnool. He is currently working as an Associate Professor & Head in Nimra College of Science & Technology the Department of Computers Science and Engineering & Technology, Jupudi, Ibrahimpatnam,Vijayawada-521456. He has more than Eight years of experience in teaching field. His area of interests are wireless networks & programming, & Mobile Computing.
E-Mail: sdyasin761@gmail.com