ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

Encrypted Cloud Data with Secure and Dynamic Multi Keyword Ranked Search Scheme

Md.Aadil Ansari[1], Sarim Shaikh[2], S.Patil[3]

Al-Ameen Educational and Medical Foundation College of Engineering, Koregaon Bhima, Pune, Maharashtra, India[1, 2]

Associate Professor, Department of Computer Engineering Al-Ameen Engineering College, Koregaon Bhima, Pune, Maharashtra, India[3]

ABSTRACT: Due to the increasing popularity of cloud computing, more and more data owners are motivated to outsource their data to cloud servers for great convenience and reduced cost in data management. However, sensitive data should be encrypted before outsourcing for privacy requirements, which obsoletes data utilization like keyword-based document retrieval. In this paper, we present a secure multi-keyword ranked search scheme over encrypted cloud data, which simultaneously supports dynamic update operations like deletion and insertion of documents. Specifically, the vector space model and the widely-used TF x IDF model are combined in the index construction and query generation. We construct a special tree-based index structure and propose a “Greedy Depth-first Search” algorithm to provide efficient multi-keyword ranked search. The secure kNN algorithm is utilized to encrypt the index and query vectors, and meanwhile ensure accurate relevance score calculation between encrypted index and query vectors. In order to resist statistical attacks, phantom terms are added to the index vector for blinding search results. Due to the use of our special tree-based index structure, the proposed scheme can achieve sub-linear search time and deal with the deletion and insertion of documents flexibly. Extensive experiments are conducted to demonstrate the efficiency of the proposed scheme.

KEYWORDS: Searchable encryption, multi-keyword ranked search, dynamic update, cloud computing,Social Networks, Collaborative, Data Sharing, Privacy Conflict, Access Control.

I. INTRODUCTION

Cloud computing has been considered as a new model of enterprise IT infrastructure, which can organize huge resource of computing, storage and applicationsCloud services, outsourcing sensitive information (such as e-mails, personal health records, company finance data, government documents, etc.) to remote servers brings privacy concern.However, this will cause a huge cost in terms of data usability. For example, the existing techniques on keyword-based information retrieval, which are widely used on the plaintext data, cannot be directly applied on the encrypted dataTherefore in this project we focus the applications of Encrypted Cloud Data with Secure and Dynamic Multi Keyword Ranked Search Scheme.

Cloud enables large group of remote servers to be in a network so as to allow the centralized data repository, and access to the computer services or resources whenever required. Many users are motivated to outsource their confidential data for example health documents, tax documents, financial documents, Emails and so on to the cloud. As the documents get transferred to the cloud, users don’t have physical possession of that data. So as to make sure that the data at cloud side is safer, we have to adapt to the privacy preserving storage, as the cloud server is not a trusted

server. To protect data confidentiality and unauthorized access to the cloud data, owners are motivated to encrypt their data and then outsource to cloud. As user is going to store encrypted data at cloud side, traditional searching will not be effective. To meet the effective searching on the encrypted cloud data, multi-keyword query should be formed and fired so as to get the top relevant data of user interest. Existing system is only working either on single query searching technique or on single user with Boolean keyword search technique. The existing searching techniques for data retrieval are only limited for single keyword queries. These searching techniques fetch all the relevant data matching the specified keyword without ranking the data of user interest.

DOI: 10.15680/IJIRCCE.2017. 0510076

II. RELATED WORK

Searchable encryption schemes enable the clients to store the encrypted data to the cloud and execute keyword search over ciphertext domain. Due to different cryptography primitives, searchable encryption schemes can be constructed using public key based cryptography or symmetric key based cryptography Song et al. proposed the first symmetric searchable encryption(SSE)scheme,andthesearchtimeoftheirscheme is linear to the size of the data collection. Gohproposed formal security definitions for SSE and designed a scheme based on Bloom filter. The search time of Goh's scheme is On , where n is the cardinality of the document collection. Curtmolaet al. proposed two schemes(SSE-1andSSE-2) which achieve the optimal search time. Their SSE-1 scheme is secure against chosen-keyword attacks (CKA1) and SSE-2 is secure against adaptive chosen-keyword attacks (CKA2). These early works are single keyword boolean search schemes, which are very simple in terms of functionality. Afterward, abundant works have been proposed under different threat models to achieve various search functionality, such as single keyword search, similarity search multi-keyword boolean search ranked search and multikeywordrankedsearch etc.

Ranked search can enable quick search of the most relevant data. Sending back only the top-k most relevant documents can effectively decrease network traffic. Some early works have realized the ranked search using order-preserving techniques, but they are designed only for single keyword search. Cao et al. realized the first privacy-preserving multi-keyword ranked search scheme, in which documents and queries are represented as vectors of dictionary size. With the "coordinate matching", the documents are ranked according to the number of matched query keywords. However, Cao et al. scheme does not consider the importance of the different keywords, and thus is not accurate enough. In addition, the search efficiency of the scheme is linear with the cardinality of document collection. Sun et al. presented a secure multi-keyword search scheme that supports similarity-based ranking. The authors constructed a searchable index tree based on vector space model and adopted cosine measure together with TFIDF to provide ranking results. Sun et al. search algorithm achieves better-than-linear search efficiency but results in precision loss. €Orencik et al. proposed a secure multikeyword search method which utilized local sensitive hash (LSH) functions to cluster the similar documents. The LSH algorithm is suitable for similar search but cannot provide exact ranking. In Zhang et al. proposed a scheme to deal with secure multi-keyword ranked search in a multiowner model. In this scheme, different data owners use different secret keys to encrypt their documents and keywords while authorized data users can query without knowing keys of these different data owners. The authors proposed an "Additive Order Preserving Function" to retrieve the most relevant search results. However, these works don't support dynamic operations.

III. ARCHITECTURE

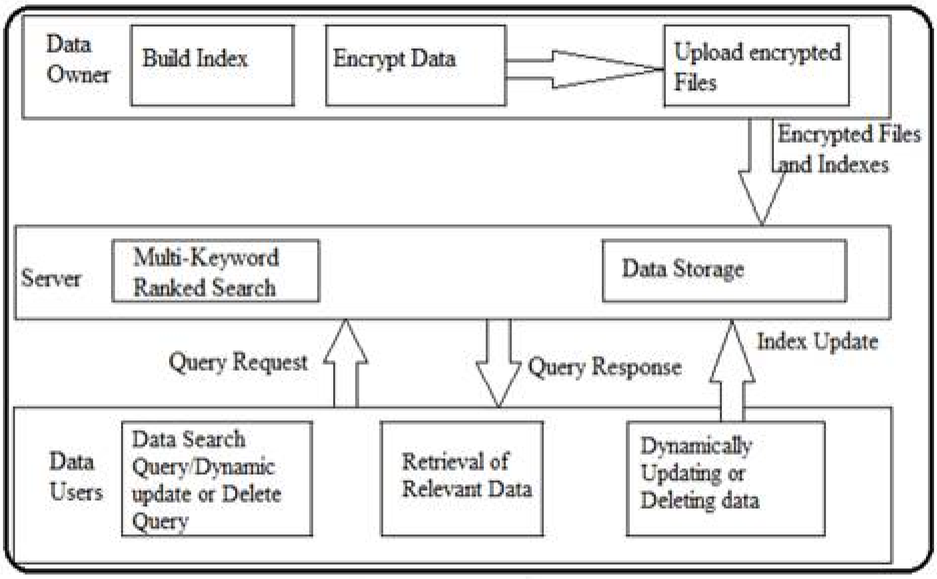

Data Owner
Build Index
Encrypt Data
Upload encrypted Files
Encrypted Files and Indexes
Server
Multi-Keyword Ranked Search
Data Storage
Query Request
Query Response
Index Update
Data Users
Data Search Query/Dynamic update or Delete Query
Retrieval of Relevant Data
Dynamically Updating or Deleting data

Fig.System Architecture

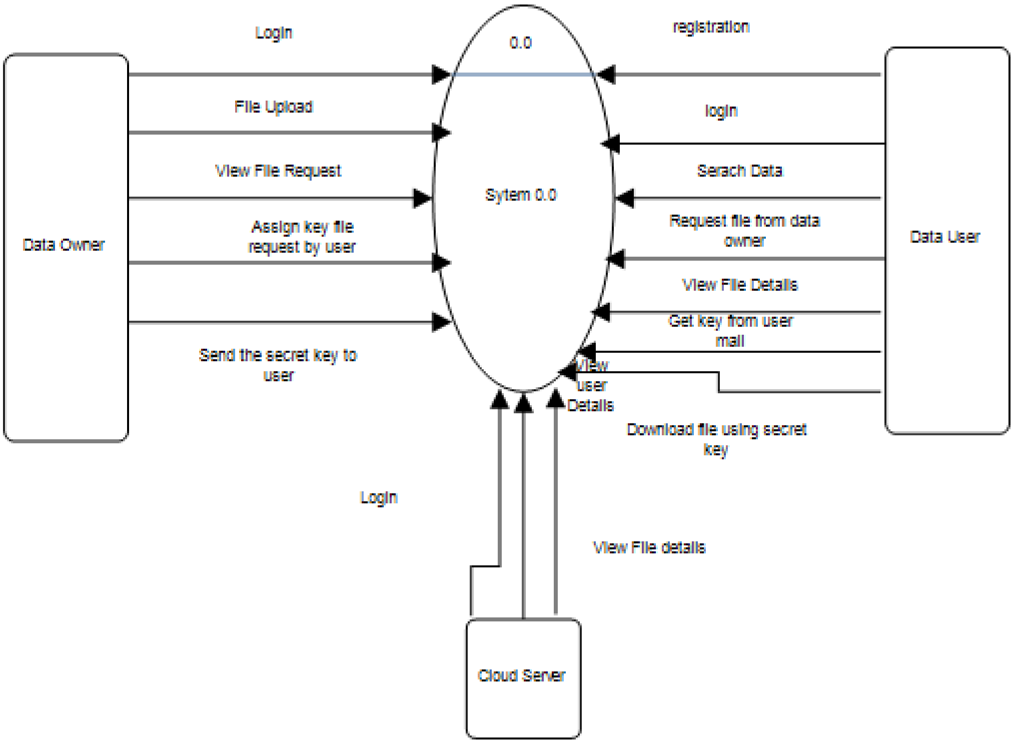


Fig.Data Flow Diagram

IV. INDEX CONSTRUCTION OF UDMRS SCHEME

In the process of index construction, we first generate a tree node for each document in the collection. These nodes are the leaf nodes of the index tree. Then, the internal tree nodes are generated based on these leaf nodes. The formal construction process of the index is presented in Algorithm 1. An example of our index tree is shown in Fig. Note that the index treeT built here is a plaintext. Following are some notations for Algorithm 1. Besides, the data structure of the tree node is defined as hID;D;Pl;P r;FIDi, where the unique identity ID for each tree node is generated through the function GenID().

- CurrentNodeSet—The set of current processing nodes which have no parents. If the number of nodes is even, the cardinality of the set is denoted as 2hðh 2 ZþÞ, else the cardinality is denoted as ð2hþ1Þ.
- TempNodeSet—The set of the newly generated nodes.

In the index, if Du½i6¼ 0 for an internal node u, there is at least one path from the node u to some leaf, which indicates a document containing the keyword wi. In addition, Du½ialways stores the biggest normalized TF value of wi among its child nodes. Thus, the possible largest relevance score of its children can be easily estimated.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

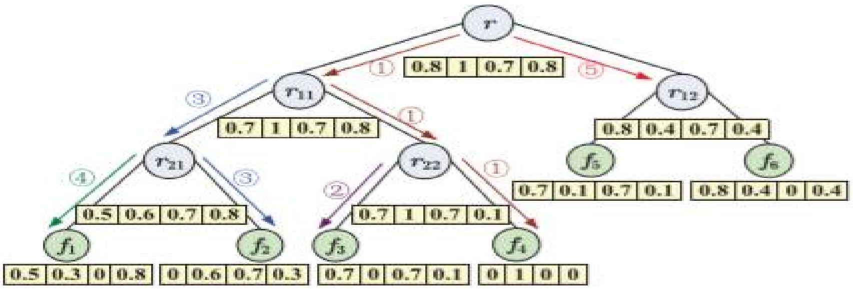
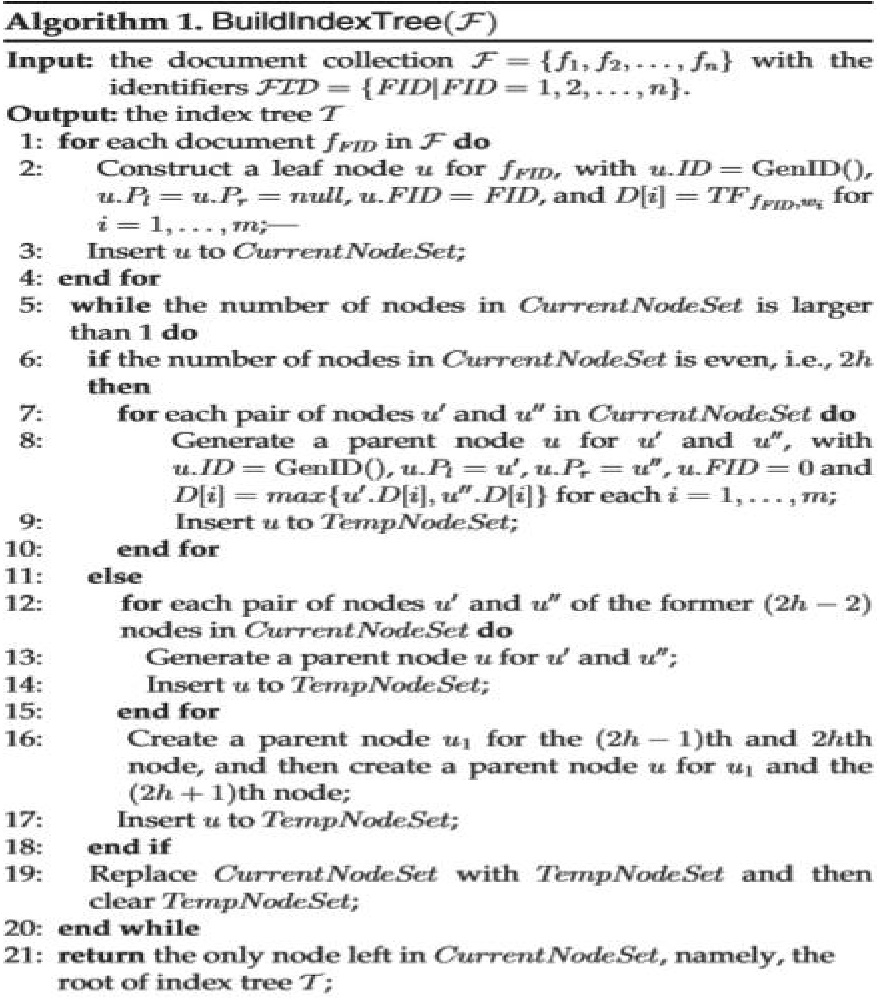


Fig.An example of the tree-based index with the document collection F¼f fiji ¼ 1;...;6g and cardinality of the dictionary m ¼ 4. In the construction process of the tree index, we first generate leaf nodes from the documents. Then, the internal tree nodes are generated based on the leafnodes.Thisfigurealsoshowsanexampleofsearchprocess,inwhich the query vector Q is equal toð0;0:92;0;0:38Þ. In this example, we set the parameter k ¼ 3 with the meaning that three documents will be returned to the user. According to the search algorithm, the search starts with the root node, and reaches the first leaf node f4 through r11 and r22. The relevance score of f4 to the query is 0:92. After that, the leaf nodes f3 and f2 are successively reached with the relevance scores 0:038 and 0:67. Next, theleafnodef1 isreachedwithscore0:58andreplacef3 inRList.Finally, the algorithm will try to search subtree rooted by r12, and find that there are no reasonable results in this subtree because the relevance score of r12 is0:52,whichis smaller thanthesmallest relevancescore inRList.



Algorithm 1. BuildIndexTree($\mathcal{F}$)

```
Input: the document collection $\mathcal{F} = \{f_1, f_2, \ldots, f_n\}$ with the
       identifiers $\mathcal{FID} = \{FID|FID = 1, 2, \ldots, n\}$.
Output: the index tree $\mathcal{T}$
for each document $f_{FID}$ in $\mathcal{F}$ do
    Construct a leaf node $u$ for $f_{FID}$, with $u.ID$ = GenID(),
    $u.P_l = u.P_r = null$, $u.FID = FID$, and $D[i] = TF_{f_{FID}, w_i}$ for
    $i = 1, \ldots, m$;—
    Insert $u$ to $CurrentNodeSet$;
end for
while the number of nodes in $CurrentNodeSet$ is larger
than 1 do
    if the number of nodes in $CurrentNodeSet$ is even, i.e., $2h$
    then
        for each pair of nodes $u'$ and $u''$ in $CurrentNodeSet$ do
            Generate a parent node $u$ for $u'$ and $u''$, with
            $u.ID$ = GenID(), $u.P_l = u'$, $u.P_r = u''$, $u.FID = 0$ and
            $D[i] = max\{u'.D[i], u''.D[i]\}$ for each $i = 1, \ldots, m$;
            Insert $u$ to $TempNodeSet$;
        end for
    else
        for each pair of nodes $u'$ and $u''$ of the former $(2h - 2)$
        nodes in $CurrentNodeSet$ do
          Generate a parent node $u$ for $u'$ and $u''$;
          Insert $u$ to $TempNodeSet$;
        end for
        Create a parent node $u_1$ for the $(2h - 1)$th and $2h$th
        node, and then create a parent node $u$ for $u_1$ and the
        $(2h + 1)$th node;
        Insert $u$ to $TempNodeSet$;
    end if
    Replace $CurrentNodeSet$ with $TempNodeSet$ and then
    clear $TempNodeSet$;
end while
return the only node left in $CurrentNodeSet$, namely, the
root of index tree $\mathcal{T}$;
```

DOI: 10.15680/IJIRCCE.2017. 0510076

AES Encryption

AES comprises three block ciphers: AES-128, AES-192 and AES-256. Each cipher encrypts and decrypts data in blocks of 128 bits using cryptographic keys of 128-, 192- and 256-bits, respectively. Symmetric (also known as secret-key) ciphers use the same key for encrypting and decrypting, so the sender and the receiver must both know -- and use -- the same secret key. All key lengths are deemed sufficient to protect classified information up to the "Secret" level with "Top Secret" information requiring either 192- or 256-bit key lengths. There are 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys -- a round consists of several processing steps that include substitution, transposition and mixing of the input plaintext and transform it into the final output of cipher text. The AES encryption algorithm defines a number of transformations that are to be performed on data stored in an array.

The first step of the cipher is to put the data into an array; after which the cipher transformations are repeated over a number of encryption rounds. The number of rounds is determined by the key length, with 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. The first transformation in the AES encryption cipher is substitution of data using a substitution table; the second transformation shifts data rows, the third mixes columns. The last transformation is a simple exclusive or (XOR) operation performed on each column using a different part of the encryption key -- longer keys need more rounds to complete.

V. CONTRIBUTION

- We suggest two MRSE schemes based on the Similarity calculation of “coordinate matching” at the time of assembling different privacy needs in two different threat models.
- We examine some further improvements of our ranked search method to maintain more search semantics and dynamic data process.
- We determine the problem of multi keyword ranked search over encrypted cloud data, and set up a set of privacy needs for such a secure cloud data operation system.
- Detailed analysis investigating privacy and Efficiency assurance of the proposed schemes is known, and testing on the real-world data set further show the proposed schemes certainly bring in low overhead on calculation and communication. In this paper we propose two new methods to maintain more search semantics. These methods also study the support of data/index dynamics in the system design.

VI. CONCLUSION AND FUTURE WORK

In this paper, a secure, efficient and dynamic search scheme is proposed, which supports not only the accurate multikeyword ranked search but also the dynamic deletion and insertion of documents. 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 byusing the secure KNN algorithm. Experimental results demonstrate the efficiency of our proposed scheme. There are still many challenge problems in symmetric SE schemes. In the proposed scheme, the data owner is responsible for generating updating information and sending them to the cloud server. Thus, the data owner needs to store the unencrypted index tree and the information that are necessary to recalculate the IDF values. Such an active data owner may not be very suitable for the cloud computing model. It could be a meaningful but difficult future work to design a dynamic searchable encryption scheme whose updating operationcanbecompletedbycloudserveronly,meanwhile reserving the ability to support multi-keyword ranked search. In addition, as the most of works about searchable encryption, our scheme mainly considers the challenge from the cloud server. Actually, there are many secure challenges in a multi-user scheme. First, all the users usually keep the same secure key for trapdoor generation in a symmetric SE scheme. In this case, the revocation of the user is big challenge. If it is needed to revoke a user in this scheme, we need to rebuild the index and distribute the new secure keys to all the authorized users. Second, symmetric SE schemes usually assumethatallthedatausersaretrustworthy.Itisnotpractical and a dishonest data user will lead to many secure problems. For example, a dishonest data user may search the documents and distribute the decrypted documents to the unauthorized ones.

DOI: 10.15680/IJIRCCE.2017. 0510076

REFERENCES

[1] E.-J. Goh, “Secure indexes,” IACR Cryptol. ePrint Archive, vol. 2003, p. 216, 2003.
[2] Y.-C. Chang and M. Mitzenmacher, “Privacy preserving keyword searches on remote encrypted data,” in Proc. 3rd Int. Conf. Appl. Cryptography Netw. Secur., 2005, pp. 442–455.
[3] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, “Searchable symmetric encryption: Improved definitions and efficient constructions,” in Proc. 13th ACM Conf. Comput. Commun. Secur., 2006, pp. 79–88.
[4] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, “Fuzzy keyword search over encrypted data in cloud computing,” in IEEE Proc. INFOCOM, 2010, pp. 1–5.
[5] M. Kuzu, M. S. Islam, and M. Kantarcioglu, “Efficient similarity search over encrypted data,” in Proc. IEEE 28th Int. Conf. Data Eng., 2012, pp. 1156–1167.
[6] C. Wang, K. Ren, S. Yu, and K. M. R. Urs, “Achieving usable and privacy-assured similarity search over outsourced cloud data,” in Proc. IEEE INFOCOM, 2012, pp. 451–459.
[7] B. Wang, S. Yu, W. Lou, and Y. T. Hou, “Privacy-preserving multikeyword fuzzy search over encrypted data in the cloud,” in Proc. IEEE INFOCOM, 2014, pp. 2112–2120.
[8] P. Golle, J. Staddon, and B. Waters, “Secure conjunctive keyword search over encrypted data,” in Proc. Appl. Cryptography Netw. Secur., 2004, pp. 31–45.
[9] Y. H. Hwang and P. J. Lee, “Public key encryption with conjunctive keyword search and its extension to a multi-user system,” in Proc. 1st Int. Conf. Pairing-Based Cryptography, 2007, pp. 2–22.
[10] L. Ballard, S. Kamara, and F. Monrose, “Achieving efficient conjunctive keyword searches over encrypted data,” in Proc. 7th Int. Conf. Inf. Commun. Secur., 2005, pp. 414–426.
[11] D. Boneh and B. Waters, “Conjunctive, subset, and range queries on encrypted data,” in Proc. 4th Conf. Theory Cryptography, 2007, pp. 535–554.
[12] B. Zhang and F. Zhang, “An efficient public key encryption with conjunctive-subset keywords search,” J. Netw. Comput. Appl., vol. 34, no. 1, pp. 262–267, 2011.
[13] J. Katz, A. Sahai, and B. Waters, “Predicate encryption supporting disjunctions, polynomial equations, and inner products,” in Proc. Adv. Cryptol., 2008, pp. 146–162.
[14] E. Shen, E. Shi, and B. Waters, “Predicate privacy in encryption systems,” in Proc. 6th Theory oCryptography Conf. Theory Cryptography., 2009, pp. 457–473.
[15] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data,” Proc. IEEE INFOCOM, pp. 829- 837, Apr, 2011.
[16] L.M. Vaquero, L. Rodero-Merino, J. Caceres, and M.Lindner, “A Break in the Clouds: Towards a Cloud Definition,” ACM SIGCOMM Comput. Commun. Rev., vol. 39, no. 1, pp. 50-55, 2009.
[17] N. Cao, S. Yu, Z. Yang, W. Lou, and Y. Hou, “LT Codes-Based Secure and Reliable Cloud Storage Service,” Proc. IEEE INFOCOM, pp. 693701, 2012. *4+ S. Kamara and K. Lauter, “Cryptographic Cloud Storage,” Proc. 14th Int’l Conf. Financial Cryptograpy and Data Security, Jan. 2010. *5+ A. Singhal, “Modern Information Retrieval: A Brief Overview,” IEEE Data Eng. Bull., vol. 24, no. 4, pp. 35- 43, Mar. 2001.
[18] I.H. Witten, A. Moffat, and T.C. Bell, Managing Gigabytes: Compressing and Indexing Documents and Images. Morgan Kaufmann Publishing, May 1999. *7+ D. Song, D. Wagner, and A. Perrig, “Practical Techniques for Searches on Encrypted Data,” Proc. IEEE Symp. Security and Privacy, 2000.
[19] E.-J. Goh, “Secure Indexes,” Cryptology ePrint Archive, http:// eprint.iacr.org/2003/216. 2003. [9] Y.-C. Chang and M. Mitzenmacher, “Privacy Preserving Keyword Searches on Remote Encrypted Data,” Proc. Third Int’l Conf. Applied Cryptography and Network Security, 2005.
[20] R. Curtmola, J.A. Garay, S. Kamara, and R. Ostrovsky, “Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions,” Proc. 13th ACM Conf. Computer and Comm. Security (CCS ’06), 2006.
[21] D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, “Public Key Encryption with Keyword Search,” Proc. Int’l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2004.

DOI: 10.15680/IJIRCCE.2017. 0510076