A Novel Approach for Secure Ranked Multi-Keyword Search Using Multiple Data Owners in Cloud Computing

Shaik Abdul Rafi#1, V. Kamakshamma

M.Tech Student 2ndyear, Dept. of CSE, PBR VITS, Kavali, India

Asst Professor, Dept. of CSE, PBRVITS, Kavali, India

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_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: secure multi-keyword ranked search.

I.INTRODUCTION

The effective data retrieval need, the large amount of documents demand the cloud server to perform result relevance ranking, instead of returning undifferentiated results. Such ranked search system enables data users to find the most relevant information quickly, rather than burdensomely sorting through every match in the content collection.

Ranked search can also elegantly eliminate unnecessary network traffic by sending back only the most relevant data, which is highly desirable in the "pay-as-youuse" cloud paradigm. For privacy protection, such ranking operation, however, should not leak any keyword related information. On the other hand, to improve the search result accuracy as Toll as to enhance the user searching experience, it is also necessary for such ranking system to support multiple keywords search, as single keyword search often yields far too coarse results.

It define and solve the problem of multi-keyword ranked search over encrypted cloud data while preserving strict system wise privacy in the cloud computing paradigm. Among various multi-keyword semantics, To choose the efficient similarity measure of "coordinate matching," i.e., as many matches as possible, to capture the relevance of data documents to the search query. Specifically, To use "inner product similarity", i.e., the number of query keywords appearing in a document, to quantitatively evaluate such similarity measure of that document to the search query.

During the index construction, each document is associated with a binary vector as a sub-index where each bit represents whether corresponding keyword is contained in the document. The search query is also described as a binary vector where each bit means whether corresponding keyword appears in this search request, so the similarity could be exactly measured by the inner product of the query vector with the data vector. However, directly outsourcing the data vector or the query vector will violate the index privacy or the search privacy.

To meet the challenge of supporting such multi keyword semantic without privacy breaches, To propose a basic idea for the PRMSM using secure inner product computation, which is adapted from a secure k-nearest neighbor (KNN) technique, and then give two significantly improved PRMSM schemes in a step-by-step manner to achieve various stringent privacy requirements.

II LITERATURE SURVEY

The encryption on data is an effective way to protect the confidentiality of data in cloud. But when it comes to searching, efficiency gets low. In literature many research works are not efficient in searching specially for complex queries. This inefficiency may lead to leakage of valuable information to unauthorized peoples. Song et al, for the first time proposed the practical symmetric searchable method based on cryptography. In this scheme the file is encrypted word by word. To search for a keyword user sends the keyword with same key to the cloud. The drawback of this scheme is that the word frequency will be revealed. Goh et al tried to overcome the drawback of Song's scheme by constructing secure index table using pseudorandom functions and unique document identifier randomized bloom filters. Bosch et al worked on the concept given by Goh et al. and introduced the concept of wild card searches. The drawback of this scheme is that bloom filters may introduce false positives. In Chang's et al proposed scheme, an index is built for each document. The scheme is more secured compared to Goh's scheme since number of words in a file is not disclosed. The limitation of this scheme is that it is less efficient and does not support arbitrary updates with new words. Golle et al scheme allows multiple keyword searches with one encrypted query. But this scheme is not practical. Although many researchers across the globe have been investigating to identify a suitable privacy preserving technique for cloud domain, none of these solutions guarantee 100 percent privacy. There exists a wide range of research challenges. We therefore chose to work towards meeting this challenge.

III PROBLEM FORMULATION

Searchable Encryption (SE) schemes maintain the confidentiality and privacy of owner's data by facilitating searching keywords directly on encrypted data. Users can upload their encrypted data to cloud. Later, the authorized users can perform private keyword search on encrypted data in cloud. Multiple domains like cryptography, indexing, storage etc. are involved in devising efficient, secure, SE algorithms over encrypted files. The participants of a secure search model in a cloud, typically involves data owner, data user and cloud server. Data owner encrypts the files and corresponding keywords based index files by using any known cryptographic algorithms. Both the encrypted files and index files are uploaded to the cloud server. The trapdoors (encrypted keywords) are used to search encrypted files by cloud server in cloud database.

A. System Model

Our system consists of 3 entities data owner, data user and the cloud server as shown in Figure 1. 1. Data owner encrypts the data files for securing the data in cloud using Commutative RSA (CRSA) before uploading into the cloud. They also define the access rights for the user who want to access those documents. The access right is a 2-state variable: permission granted or permission denied. Data owner creates an index tree based on B tree and encrypts the tree using CRSA.

2. Cloud server stores the encrypted data files and encrypted index tree. It accepts the encrypted keywords (trapdoor) and returns the matching data file based on their relevance.

3. Data user can search for encrypted data files in cloud with encrypted keywords (trapdoor). The purpose of using encrypted keywords is that even the cloud server must not be able to infer the contents of data files.

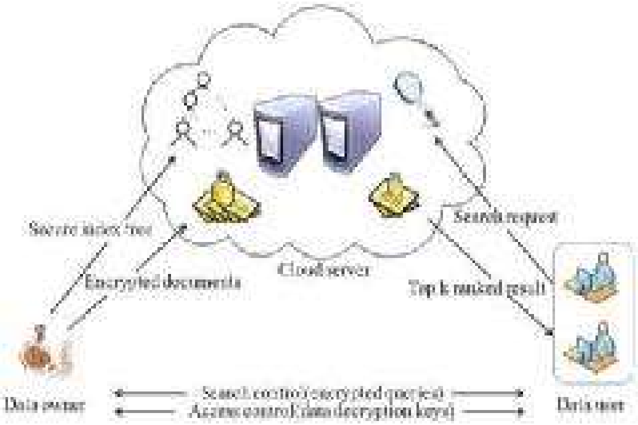

Figure 1: Searchable Encryption Architecture using CRSA

B. Threat Model

The threat model for our search scheme adopts "honestbut- curious" cloud server, that is the cloud server "honestly" follows the protocol specification, but it is "curious" to infer and analyze data (including indexes) in its storage and message flows received during the protocol in order to learn additional information.

C. Design Goals

The proposed solution addresses the following requirements

1. The search on encrypted document/file must be fully secure and cloud server must not be able to infer the contents of the documents in any way.

The search results must be ranked in order of relevance To enable ranked searchable encryption for effective utilization of outsourced and encrypted cloud data under the aforementioned model, our system design should achieve the following security and performance guarantee. Specifically, we have the following goals: 1) Ranked keyword search: to explore different mechanisms for designing effective ranked search schemes based on the existing searchable encryption framework; 2) Security guarantee: to prevent cloud server from learning the plaintext of either the data files or the searched keywords, and achieve the "as-strong-as-possible" security strength compared to existing searchable encryption schemes; 3) Efficiency: above goals should be achieved with minimum communication and computation overhead.

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. The mathematical scheme for performing this encryption is described by a pseudo algorithm presented below.

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.

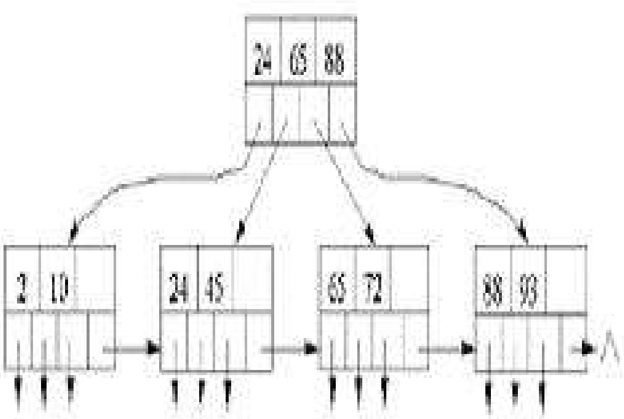

Figure 2: B tree data structure

IV SEARCHABLE ENCRYPTION SCHEME

To design an efficient multi-keyword searchable encryption scheme based on public key cryptography, we included the following modules.

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. 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.

Index Module: Index structures for huge datasets cannot be stored in main memory. Disk is a possible alternative. Storing it on disk requires different approach. The solution is to use more branches to reduce the height of the tree. For this we used B-tree data structure for each document. B-tree is a data structure of order n. The nodes are filled from n to 2n keys. Nodes are always at least half full of keys. The keys are within each node. A list of pointers is inserted between keys. These pointers help to navigate through tree. In general, a node with k keys has (k+1) pointers.

V CONCLUSION AND FUTURE WORK

This work uses CRSA asymmetric algorithm for encrypting data files and index tree based on B-tree. CRSA increases the data security and improves privacy of data by its commutative nature. 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.

REFERENCES

[1] M. Armbrust et al., ‘Above the Clouds: A Berkeley View of Cloud Computing,’ Feb 2009.

[2] S. Kamara and K. Lauter, ‘Cryptographic cloud storage,’ in RLCPS, January 2010, LNCS. Springer, Heidelberg.

[3] A. Singhal, ‘Modern information retrieval: A brief overview,’ IEEE Data Engineering Bulletin, vol. 24, no. 4, pp. 35–43, 2001.

[4] Cloud Security Alliance, ‘Security Guidance for Critical Areas of Focus in Cloud Computing,’ http://www.cloudsecurityalliance.org, 2009.

[5] R. Brinkman, ‘Searching in encrypted data,’ in University of Twente, PhD thesis, 2007.

[6] Ning Cao; Cong Wang; Ming Li; Kui Ren; Wenjing Lou, ‘Privacy- Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data,’ Parallel and Distributed Systems, IEEE Transactions on , vol.25, no.1, pp.222,233, Jan. 2014

[7] Dawn Xiaoding Song; Wagner, D.; Perrig, A., ‘Practical techniques for searches on encrypted data,’ Security and Privacy, 2000.S&P 2000. Proceedings. 2000 IEEE Symposium on ,doi: 10.1109/SECPRI.2000.848445 vol., no., pp.44,55, 2000

[8] J. Li et al., ‘Fuzzy Keyword Search Over

Encrypted Data in Cloud Computing,’ Proc. IEEE INFOCOM ’10 Mini-Conf., San Diego, CA, Mar. 2010.

[9] M. Li et al., ‘Authorized Private Keyword Search over Encrypted Data in Cloud Computing,’ 31st Int’l. Conf. Distributed Computing Systems, 2011, pp. 383–92.

[10] D. 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 EncryptionandDistance-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.