Cloud Data Encryption Using RSA,Enabling Multi-Keyword Ranked Search and Achieving Privacy Requirements

Akshatha MS[1], Renita Tellis[2]

Student, Computer Science & Engineering, MITE, Moodabidri, India[1]

Asst. Professor, Computer Science & Engineering, MITE, Moodabidri, India [2]

Abstract: Cloud computing, is also known as on-demand computing, and provides services to customers over the internet. It serves users by allowing them to store and access data from cloud server at anytime, anywhere, using any devices. Users upload their private data on to the cloud which has to be kept a secret. For such secure information storage many go for encryption approach before outsourcing to cloud. An efficient search scheme needs to be used in order to search data over the encrypted cloud. While having a large number of data users, and documents on the cloud, it is essential to give multi-keywords in the search query to get documents only relevant to these keywords in a ranked order. Coordinate matching and inner product similarity are the multi-keyword concepts used to get the relation between the documents and search queries and similarity measures, respectively.

Keywords: Cloud computing, multi-keyword, data privacy, inner product similarity, coordinate matching.

I. INTRODUCTION

Nowadays due to large amount of information flowing continuously through and around organizations, it has become a tedious work for IT practitioner to manage everything including new hardware and software. This problem can be overcome to some extent by shifting to a centralized computing paradigm called as Cloud computing. Cloud computing is a shared pool of computing resources to store or access data from a remote place. It includes moving enterprises' or individuals' database and applications to servers which will be managed by the third-party cloud service providers (CSP) [1]. People find cloud computing interesting due to its benefits like rapid elasticity, on-demand services, resource pooling, pay as you use and broad network access etc.

In spite of all the above advantages, cloud computing still has some issues needed to be handled efficiently. The major two issues to be handled are security and privacy. Privacy is the obstruction that forbids the widespread adoption of the cloud by many users. A cloud service provider has the entire control over cloud infrastructure. Although sensitive data can be protected by using firewalls, intrusion detection system, etc., privacy is still fully unattainable because of the existence of dissatisfied or curious employees present in the CSP.

Encrypting the data before outsourcing to the server actually provides protection thereby preventing intruders from getting any useful information from the ciphertext of users' data. Many people advise that sensitive data needs encryption before outsourcing in order to provide user data privacy against the cloud service providers also. Nevertheless, encrypted data on the cloud makes data utilization a very challenging task. For example, the Plain keyword search functions cannot be used to retrieve the encrypted documents stored over the cloud. Hence limiting the data services, the cloud will become simply a remote storage that offers limited value to the parties.

Computation over encrypted data is a hard task and has drawn substantial attention due to the encrypting then outsourcing paradigm in cloud computing. It would be negligent that if not mentioned, the fully homomorphic encryption [2], which is considered as the Holy Grail of cryptography. The fully homomorphic encryption scheme will allow operating directly over ciphertext and generating results matching the computation over plain text. A theoretical break-through on fully homomorphic encryption took place many years ago.

However, the efficiency in the construction is still far from being practical. Search over encrypted data is a basic and usual form of data utilization service, which makes users to quickly obtain information of interest from large amount of data, and thus has become a matter of interest recently.

Both public key cryptography (PKC) and symmetric key cryptography (SKC) can be used to build encrypted data search schemes[1].On the whole, PKC-based schemes [3]are more expressive and also supports more flexible search functions, but more computationally hard, whereas SKC-based schemes [4] are more potent in searching, but less flexible in the search criteria backed.

The main aim is to enable a scheme that achieves privacy preserving multi-keyword search which supports similarity based ranking.

II. LITERATURE SURVEY

A. Providing privacy preserving in cloud computing

Privacy is a major issue in cloud computing, user trust needs to be taken into consideration during all phases of the plan. The [5] paper is about the Importance of protecting individual's data privacy in cloud computing and gives some technologies which preservers privacy in cloud computing services. There are some drawbacks in this proposed system, the paper only describes about privacy of data but does not include anything about indexed search and also does not hide users' identity.

B. Public key encryption with keyword search

Dan Boneh gave an idea for searching over the cloud data and the solution is encrypted using the Public key Crypto System [6]. The main idea is to securely attach or tag the related keywords of each document along with the file. This will avoid the necessity of decrypting the files entirely and by saving the time for scanning the whole file to check for the existence of keywords. Here, the file is encrypted using an algorithm that uses a public key encryption [6] and having keyword W, sending only the Trapdoor (W) to the server. The two methods for construction of the scheme have been proposed, one with the use of bilinear maps and other using Jacobi symbols. The problem with this approach is that every tag of all the files has to be processed in order to find the matches.

C. Boolean symmetric searchable encryption:

The techniques discussed so far focused mainly on single keyword matching but in practice users wants to enter more than one keyword so as to get relevant data accurately. Tarik Moataz introduced a solution to handle challenges regarding search using multiple keywords over the encrypted cloud data. The construction of Boolean Symmetric Searchable Encryption (BSSE) [7] is basically using orthogonalisation of the keyword field according to the Gram-Schmidt process. The basic Boolean operations used here are: the disjunction, the conjunction and the negation. The conjunction search incurs large overhead whereas disjunction search gives undifferentiated results.

D. Fuzzy Keyword Search:

Traditional searching techniques for retrieving files only based upon exact keyword match, but the Fuzzy keyword search technique further extends by supporting common literal errors and format incompatibilities which would happen when the user types the keywords. The data privacy is retained during an exact keyword search is guaranteed only by making use of this proposed method. Wild card based technique [8] is used to build efficient fuzzy keyword sets that are used for matching appropriate documents. The keyword sets are created using an Edit Distance algorithm which quantifies word similarity. Here, the keyword sets reduce the storage and representation overhead by removing the need to generate all fuzzy keywords, rather generating on similarity basis. The search result that is given back when search based on a fuzzy keyword data set is generated at every time the exact match search fails.

III. PROBLEM FORMULATION

A. System Design

Consider a cloud data hosting service linking three distinct individuals, as shown in Fig. 1: the data owner, the data user, and the cloud server. The data owner has a set of data documents F that needs to be outsourced in the encrypted form C. In order to provide the good searching capability over C, the data owner must build an encrypted searchable index I from F prior to outsourcing the documents. While searching a file with the given t keywords, only authorized users having secret key can download files. Once the keyword query is given to cloud server, cloud server has the responsibility of searching file for index I, and returning the only relevant files to keywords. The files are displayed in their ranked relevance to keywords. Finally, to manage the file downloading task, access control mechanism is employed.

B. Notations

- F—the file collection, represented as a set of m files F= $(F_1, F_2 \ldots F_m)$.
- C—the file collection which are encrypted and stored in the cloud server, represented as C= $(C_1, C_2 \ldots C_m)$.
- W—the keyword dictionary, represented as W= $(W_1, W_2 \ldots W_m)$.
- I—the searchable index related with C, represented as $(I_1,I_2 \ldots I_m)$.where each sub index I_i is constructed for F_i.
- $\widetilde{W}$-the subset of W, representing the keywords in a search request, denoted as $\widetilde{W}=(\widetilde{W}_{j1},\widetilde{W}_{j2} \ldots \widetilde{W}_{jt})$
- $T\widetilde{W}$—the trapdoor for the search request $\widetilde{W}$.
- $F\widetilde{W}$ —the ranked list of all files according to their relevance to $\widetilde{W}$.

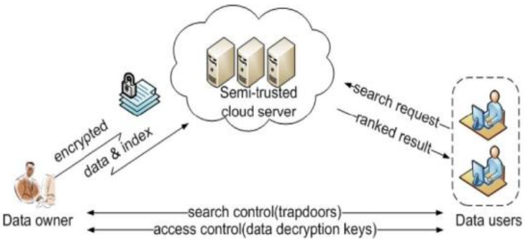
Semi-trusted cloud server
search request
ranked result
encrypted data & index
Data owner
Data users
search control(trapdoors)
access control(data decryption keys)

Fig. 1. Architeture of search over cloud data.

C. Goals

To facilitate the multi-keyword ranked search for efficient retrieval of outsourced cloud data, the system design should accomplish security and performance as listed.

- Multi-keyword ranked search: To enable ranked search for successful utilization of outsourced cloud data, scheme should allow multi-keyword query and must result the similarity based ranking for effective data retrieval.
- Privacy-preserving: Prohibiting the server from learning additional information from the data set and the index.
- Access Control: Managing users' access on outsourced data using bloom filter.

DOI 10.17148/IJARCCE.2016.5587

- Data Integrity Check: Verifying the integrity of data using cryptographic hash function.

IV. FRAMEWORK FOR MRSE

The section describes the framework of searching encrypted cloud data using multi-keyword ranked search (MRSE) technique and gives brief idea about different methods involved in order to have a safe cloud data utilization system.

A. MRSE Outline

The data owner in the framework employs the traditional asymmetric key cryptography (RSA) to encrypt and then outsources data. With an attention to index and query, the MRSE system comprises following algorithms as takes after:

- Key Setup (1^l) -Considering security parameter l as input, the data owner gives an asymmetric key as K.
- Build Index (F, K) – The data set as F, the data owner constructs a searchable index I and its hash values are calculated, later outsourced to the cloud server. After constructing the index, the document collection can be separately encrypted and outsourced.
- Trapdoor ($\widetilde{W}$) - For t keywords of choice in $\widetilde{W}$ sent as input, the algorithm gives back a corresponding search query as $T\widetilde{W}$.
- Query ($T\widetilde{W}$,k,I)- Having ($T\widetilde{W}$, k) as request to cloud server, a search on index I is conducted. With making use of trapdoor $T\widetilde{W}$, the result $F\widetilde{W}$ is generated and top-k documents are ranked by their similarity with $\widetilde{W}$.
- The access control is in the scope of this paper, which is to manage users' access to the outsourced documents and integrity check for the file content is enabled for security issues.

B. Coordinate Matching

Coordinate Matching, is a similarity measure which has been integrated into the MRSE schemes in order to support multi-keyword ranked search. The measure quantifies the relevance of the document to the query by counting the query keywords appearing in the documents. Any document that consists of all keywords or only few keywords matching will be included in the result set. In practice, to formalize such similarity measure an inner product of the query vector is used. For example, assuming a dictionary having the following keyword set {server, cloud, product, networking, secure}. A and B are the two documents present in the dataset. The index vector as a binary vector D_A = (1, 0, 0, 1, 1) for document A if it contains keywords {server, networking, secure} only, where 1 indicates the existence of the keyword in the document and 0 otherwise. If the document B has got keywords {server, cloud, secure}, then the binary index vector D_B is said to be (1, 1, 0, 0, 1). If the data user has a query keywords {server, cloud, product}, then binary query vector Q would be represented as (1, 1, 1, 0, 0).Now, calculating the inner products of the query vector Q and the index vectors D_A,D_B as the similarity scores of documents A and B:

$\text{SimilarityScore}_A = Q\cdot D_A = (1, 1, 1, 0, 0) \cdot (1, 0, 0, 1, 1) = 1,$
$\text{SimilarityScore}_B = Q\cdot D_B = (1, 1, 1, 0, 0) \cdot (1, 1, 0, 0, 1) =2$

Therefore, concluding that the similarity score of B is greater than that of A and it also yields a ranking B > A. So the data user must prefer document B to document A. Hence, the coordinate matching similarity measure is an effective multi-keyword ranked search function.

C. Inner Product Evaluation

In the above mentioned similarity measure while preserving privacy, D_d -index vector for each document d, query vector Q and their inner product $D_d\cdot Q$ must not be exposed to the cloud server. In MRSE, a secure inner product scheme is proposed which is taken from a secure k-nearest neighbor (kNN) technique [10] to hide the sensitivity of information. Some modifications on the data structure needs to be done in order to fit to the MRSE framework. One way of doing that is by removing the dimension extension, the final result leads to be the inner product as $rp_i.q$. For the encryption of data record or query vector, the two multiplications of a dxd matrix and a d-dimension vector with complexity $O(d^2)$ is needed, two multiplications of two d-dimension vectors with complexity O(d) will be the final inner product computation performed. The $\overrightarrow{Pi}'$, $\overrightarrow{Pi}''$ are taken as two random d-dimensional vectors since S- the splitting factor is unknown in the ciphertext model. To resolve the linear equations build by the encryption of data vectors, in m data vectors only 2dm unknowns and $2d^2$ unknowns in (M_1,M_2) are present. Because only 2dm equations are present, there is no sufficient information in order to solve either data vectors or (M_1,M_2) since there are less than the number of unknowns. Similarly, $\overrightarrow{qi}'$, $\overrightarrow{qi}''$ are considered as two random d-dimensional vectors. To solve the linear equations produced by the query vectors encryption, only 2d unknowns in two query vectors and $2d_2$ unknowns in (M_1,M_2) are present. Since only 2d equations are present that are again less than the number of unknowns, there exists no adequate information to solve either query vectors or (M_1,M_2).

Hence, without the knowledge of secret key in advance, the analysing corresponding ciphertext is difficult. Neither the data vector nor the query vector can be found after such sequential processes of splitting and multiplication.

D. Bloom filter for controlling the users' access on cloud data. In 1970, Burton Howard Bloom proposed a space-efficient probabilistic data structure which is called as Bloom filter [12] and used to test whether an element is a member of a set. In this data structure, false positive matches are possible, but not the false negatives. Hence, a Bloom filter is said to have 100% recall rate means a query returns either "possibly in set" or "definitely not in set" only. Basic operations of bloom filter are test and add. Test is to check whether a given element is in the set or not. If the result returned is:

- false, the element is definitely not in the set.
- true, the element is probably in the set.

DOI 10.17148/IJARCCE.2016.5587

Bloom filter consists of a bit vector of length m. While adding an item to the bloom filter, k different hash functions are used or the different k is used and setting the bits at the resulting positions. Sometimes the hash functions produce overlapping positions, so setting less than k positions may be adopted. To test if an item is in the filter, again feed it to the k hash functions. Now, checking if any of the bits at these positions are not set, if any are not set, then the item is definitely not present in the set. Otherwise, probably item is in the set.

E. Cryptographic hash function for integrity check

A cryptographic hash function, which is a hash function considered as impossible to invert practically, means using its hash value alone and rebuilding the input data. The message and message digest are input data and its output (the hash value) respectively. Verifying the integrity of files or messages [11]: Checking the integrity of message is an important application of secure hashes. It is very useful in determining any changes have been made to a message (or a file), for example, it can be verified by comparing message digests that is calculated before, and after, outsourced to cloud server. If the authenticity of a hashed digest of the message is verified then it is considered as a proof that the message itself is authentic one.

V. CONCLUSION

In the paper, enabling the multi-keyword ranking search for outsourced encrypted cloud data is being achieved. The scheme makes indexes that support the user to search for his documents in a secure environment. The files matching the every keyword in the search query are further ranked based on the relevant score calculated using term frequency, file length, etc. This solves the problem of multi-keyword ranked search on encrypted cloud data, and sets a variety of privacy requirements. Among different multi-keyword semantics, by choosing the principle of “coordinate matching” to effectively represent similarity between query keywords and outsourced documents, and using “inner product similarity” the principle of similarity measurement. Studying some further enhancements of the ranked search mechanism, supports more search semantics, i.e., TF _ IDF, and dynamic data operations. Access control for the different users has been achieved and the integrity check of data which is been outsourced to the cloud has examined efficiently. In the future work, exploring the ways for integrity check of the rank order in the search result assuming the cloud server is untrusted and providing virtualization security that tries securing the execution environment on cloud server.

ACKNOWLEDGMENT

Dedicating the paper work to the esteemed guide, Asst. **Prof. Renita Tellis,** whose interest and guidance helped in the completion of work successfully. And also extending the gratitude to **Prof. Dr. Nagesh H.R** (H.O.D Computer Science & Engineering Department) who has facilitated exploring the subject with more enthusiasm.

REFERENCES

[1]. Wenhai Sun, Wenjing Lou, Y. Thomas Hou, and Hui Li, “Privacy-Preserving Keyword Search Over Encrypted Data in Cloud Computing”, Springer Science+Business Media New York 2014

[2]. Gentry, C.: A fully homomorphic encryption scheme. Ph.D. thesis, Stanford University (2009)

[3]. Boneh, D., Di Crescenzo, G., Ostrovsky, R., Persiano, G.: Public key encryption with keyword search. In: Advances in Cryptology-Eurocrypt 2004, pp. 506–522. Springer (2004)

[4]. Curtmola.R,Garay.J,Kamara.S,Ostrovsky.R.:Searchable Symmetric encryption:improved definitions and efficient constructions.In:proceedings of the 13th ACM conference on computer and communications security, pp.79-88 ACM(2006)

[5]. Shiba Sampat Kale et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (6) , 2014, 7093-7096

[6]. Ning Cao,Cong Wang, Ming Li, Member, and Wenjing Lou,“Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data” IEEE Transaction Parallel AND Distributed Ssystems, vol. 25, no. 1, january 2014

[7]. Tarik Moataz, Abdull Shikfa, “Boolean Symmetric Searchable Encryption”, Telecom Bretange,Bell Labs Research France.

[8]. Dinesh Nepolean, I.Karthik, Mu.Preeti, Rahul Goyal & M.K Vanethi, “Privacy Preserving Ranked Keyword Search over Encrypted Cloud data” Vol4,no.11, nov2013

[9]. R. Brinkman, “Searching in encrypted data”, PhD thesis, Univ. of Twente 2007

[10]. Wong, W.K., Cheung, D.W.I, Kao, B., Mamoulis, N.: Secure knn Computation on encrypted databases. In:Proceedings of the 2009 ACM SIGMOD International Conference on Mangement of data,pp.139-152. ACM(2009)

[11]. http://en.wikipedia.org/wiki/Cryptographic_hash_function

[12]. http://en.wikipedia.org/wiki/Bloom_filter

DOI 10.17148/IJARCCE.2016.5587