A Secure and Dynamic Multi-keyword Ranked Search Scheme over Encrypted Cloud Data

M.MOHANRAO, GANGAM DIVYA
Assistant Professor, M.TECH STUDENT
Dept of CSE,Megha Institute of Engineering & Technology For womens,Edulabad,Ghatkesar mandal,RangaReddy Dist,Telangana ,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

Cloud computing has been considered as a new model of enterprise IT infrastructure, which can organize huge resource of computing, storage and applications, and enable users to enjoy ubiquitous, convenient and on-demand network access to a shared pool of configurable computing resources with great efficiency and minimal economic overhead. Attracted by these appealing features, both individuals and enterprises are motivated to outsource their data to the cloud, instead of purchasing software and hardware to manage the data themselves. Despite of the various advantages of cloud services, outsourcing sensitive information (such as e-mails, personal health records, company finance data, government documents, etc.) to remote servers brings privacy concerns. The cloud service providers (CSPs) that keep the data for users may access users' sensitive information without authorization. A general approach to protect the data confidentiality is to encrypt the data before outsourcing. 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 data. Downloading all the data from the cloud and decrypt locally is obviously impractical. In order to address the above problem, researchers have designed some general-purpose solutions with fully-homomorphic encryption [3] or

oblivious RAMs [4]. However, these methods are not practical due to their high computational overhead for both the cloud sever and user. On the contrary, more practical special purpose solutions, such as searchable encryption (SE) schemes have made specific contributions in terms of efficiency, functionality and security. Searchable encryption schemes enable the client to store the encrypted data to the cloud and execute keyword search over ciphertext domain. So far, 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, multi-keyword ranked search, etc. Among them, multikeyword ranked search achieves more and more attention for its practical applicability. Recently, some *dynamic* schemes have been proposed to support inserting and deleting operations on document collection. These are significant works as it is highly possible that the data owners need to update their data on the cloud server. But few of the dynamic schemes support efficient multikeyword ranked search.

II. RELATED WORK

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, and the search time of their scheme is linear to the size of the data collection. Goh [8] proposed formal security definitions for SSE and designed a scheme based on Bloom filter. The search time of Goh's scheme is $O(n)$, where n is the cardinality of the document collection. Curtmola *et al.* [10] proposed two schemes (SSE-1 and SSE-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 ,multi-keyword boolean search, ranked search, and multi-keyword ranked search etc. Multi-keyword boolean search allows the users to input multiple query keywords to request suitable documents. Among these works, conjunctive keyword search schemes only return the documents that contain all of the query keywords. Disjunctive keyword search schemes return all of the documents that contain a subset of the query keywords. Predicate search schemes are proposed to support both conjunctive and disjunctive search. All these multikeyword search schemes retrieve search results based on the existence of keywords, which cannot provide acceptable result ranking functionality. 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.*'s 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 TF×IDF

to provide ranking results. Sun *et al.*'s search algorithm achieves better-than-linear search efficiency but results in precision loss. O¨ rencik *et al.* proposed a secure multi-keyword 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 multi-owner 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. PROBLEM STATEMENT

A. Exiting Model

A general approach to protect the data confidentiality is to encrypt the data before outsourcing. Searchable encryption schemes enable the client to store the encrypted data to the cloud and execute keyword search over ciphertext domain. So far, 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, multi-keyword ranked search, etc. Among them, multi-keyword ranked search achieves more and more attention for its practical applicability. Recently, some *dynamic* schemes have been proposed to support inserting and deleting operations on document collection. These are significant works as it is highly possible that the data owners need to update their data on the cloud server.

Disadvantages:

- 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 data. Downloading all the data from the cloud and decrypt locally is obviously impractical.
- Existing System methods not practical due to their high computational overhead for both the cloud sever and user.

B .Proposed Model

- This paper proposes a secure tree-based search scheme over the encrypted cloud data, which supports multi-keyword ranked search and dynamic operation on the document collection. Specifically, the vector space model and the widely-used "term frequency (TF) × inverse document frequency (IDF)" model are combined in the index construction and query generation to provide multi-keyword ranked search. In order to obtain high search efficiency, we construct a tree-based index structure and propose a "Greedy Depth-first Search" algorithm based on this index tree.

- 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.

- To resist different attacks in different threat models, we construct two secure search schemes: the basic dynamic multi-keyword ranked search (BDMRS) scheme in the known ciphertext model, and the enhanced dynamic multi-keyword ranked search (EDMRS) scheme in the known background model.

Advantages

Due to the special structure of our tree-based index, the proposed search scheme can flexibly achieve sub-linear search time and deal with the deletion and insertion of documents.

We design a searchable encryption scheme that supports both the accurate multi-keyword ranked search and flexible dynamic operation on document collection. Due to the special structure of our tree-based index, the search complexity of the proposed scheme is fundamentally kept to logarithmic. And in practice, the proposed scheme can achieve higher search efficiency by executing our "Greedy Depth-first Search" algorithm. Moreover, parallel search can be flexibly performed to further reduce the time cost of search process.

IV. PROBLEM FORMULATION

A. Notations and Preliminaries

- $\mathcal{W}$ – The dictionary, namely, the set of keywords, denoted as $\mathcal{W} = \{w_1, w_2, ..., w_m\}$.
- m – The total number of keywords in $\mathcal{W}$.
- $\mathcal{W}_q$ – The subset of $\mathcal{W}$, representing the keywords in the query.
- $\mathcal{F}$ – The plaintext document collection, denoted as a collection of n documents $\mathcal{F} = \{f_1, f_2, ..., f_n\}$. Each document f in the collection can be considered as a sequence of keywords.
- n – The total number of documents in $\mathcal{F}$.
- $\mathcal{C}$ – The encrypted document collection stored in the cloud server, denoted as $\mathcal{C} = \{c_1, c_2, ..., c_n\}$.
- $\mathcal{T}$ – The unencrypted form of index tree for the whole document collection $\mathcal{F}$.
- $\mathcal{I}$ – The searchable encrypted tree index generated from $\mathcal{T}$.
- Q – The query vector for keyword set $\mathcal{W}_q$.
- TD – The encrypted form of Q, which is named as trapdoor for the search request.
- D_u – The index vector stored in tree node u whose dimension equals to the cardinality of the dictionary $\mathcal{W}$. Note that the node u can be either a leaf node or an internal node of the tree.
- I_u – The encrypted form of D_u.

The system model in this paper involves three different entities: data owner, data user and cloud server, asillustrated in Fig. 1. **Data owner** has a collection of documents *F = {f1; f2; :::; fn}* that he wants to outsource to the cloud server in encrypted form while still keeping the capability to search on them for effective utilization.

In our scheme, the data owner firstly builds a secure searchable tree index *I* from document collection *F*, and then generates an encrypted docollection *C* for *F*. Afterwards, the data owner outsources the encrypted collection *C* and the secure index *I* to the cloud server, and securely distributes the key information of trapdoor generation (including keyword IDF values) and document decryption to the authorized data users. Besides, the data owner is responsible for the update operation of his documents stored in the cloud server. While updating, the data owner generates the update information locally and sends it to the server. *Data users* are authorized ones to access the documents of data owner. With *t* query keywords, the authorized user can generate a trapdoor *TD* according to search control mechanisms to fetch *k* encrypted documents from cloud server. Then, the data user can decrypt the documents with the shared secret key. *Cloud server* stores the encrypted document collection *C* and the encrypted searchable tree index *I* for dataowner. Upon receiving the trapdoor *TD* from the data user, the cloud server executes search over the index tree *I*, and finally returns the corresponding collection of top- *k* ranked encrypted documents. Besides, upon receiving the update information from the data owner, the server needs to update the index *I* and document collection *C* according to the received information. **V. DESIGN GOALS** To enable secure, efficient, accurate and dynamic multi data under the above models, our system has the following *Dynamic:* The proposed scheme is designed to provide not only multi-keyword query and accurate result ranking, but also dynamic update on document collections. *Search Efficiency:* The scheme aims to achieve sublinear search efficiency by exploring a special tree-based index and an efficient search algorithm. *A.Privacy-preserving:* The scheme is designed to prevent the cloud server from learning additional information about the document collection, the index tree, and the query. The specific privacy requirements are summarized as follows, *B. Index Confidentiality and Query Confidentiality:* The underlying plaintext information, including keywords in the index and query, TF values of keywords stored in the index, and IDF values of query keywords, should be protected from cloud server; *C. Trapdoor Unlinkability:* The cloud server should not be able to determine whether two encrypted queries (trapdoors) are generated from the

same search request; *D. Keyword Privacy:* The cloud server could not identify the specific keyword in query, index or document collection by analyzing the statistical information like term frequency. Note that our proposed scheme is not designed to protect access pattern, i.e., the sequence of returned documents.

V. CONCLUSION

In this paper, a secure, efficient and dynamic search scheme is proposed, which supports not only the accurate multi-keyword 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 by using 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 operation can be completed by cloud server only, 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. Firstly, 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. Secondly, symmetric SE schemes usually assume that all the data users are trustworthy. It is not practical 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. Even more, a dishonest data user may distribute his/her secure keys to the unauthorized ones. In the future works, we will try to improve the SE scheme to handle these challenge problems.

REFERENCES

[1] K. Ren, C.Wang, Q.Wang et al., "Security challenges for the public cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69–73, 2012.

[2] S. Kamara and K. Lauter, "Cryptographic cloud storage," in Financial Cryptography and Data Security. Springer, 2010, pp. 136–149.

[3] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Stanford University, 2009.

[4] O. Goldreich and R. Ostrovsky, "Software protection and simulation on oblivious rams," Journal of the ACM (JACM), vol. 43, no. 3, pp. 431–473, 1996.

[5] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Advances in Cryptology- Eurocrypt 2004. Springer, 2004, pp. 506–522.

[6] D. Boneh, E. Kushilevitz, R. Ostrovsky, and W. E. Skeith III, "Public key encryption that allows pir queries," in Advances in Cryptology-CRYPTO 2007. Springer, 2007, pp. 50–67.

[7] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on. IEEE, 2000, pp. 44– 55.

[8] E.-J. Goh et al., "Secure indexes." IACR Cryptology ePrint Archive, vol. 2003, p. 216, 2003.

[9] Y.-C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in Proceedings of the Third international conference on Applied Cryptography and Network Security. Springer-Verlag, 2005, pp. 442–455.

[10] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," in Proceedings of the 13th ACM conference on Computer and communications security. ACM, 2006, pp. 79–88. [11] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy keyword search over encrypted data in cloud computing," in INFOCOM, 2010 Proceedings IEEE. IEEE, 2010, pp. 1–5.

[11] M. Kuzu, M. S. Islam, and M. Kantarcioglu, "Efficient similarity search over encrypted data," in Data Engineering (ICDE), 2012 IEEE 28th International Conference on. IEEE, 2012, pp. 1156–1167.