ISSN 2395-1621

Efficient Ranked Multi-keyword Search for Encrypted Cloud with Provision of Equivalent Query

#1Rucha Shinde, #2Hema Kumbhar

1ruchashinde14@gmail.com
2kumbhar.hema344@gmail.com

#12Department of Computer Engineering, PVPIT, Bavdhan, Pune.

ABSTRACT

Now a days, outsourcing the data for storage purpose is becoming a general scenario. To serve this objective emerging cloud computing becomes most popular which provides efficient and flexible storage with reduced cost and utility of on-demand high quality applications and services. So internet usage strongly relies on cloud for privacy preserving and fast data retrieval. For consumers, they want to find the most relevant products or data, which is highly desirable in the "pay-as-you use" cloud computing paradigm. Sensitive data (such as photos, mails, health records, financial records, etc) is encrypted before outsourced to cloud. Although Searchable encryption scheme has been developed to conduct retrieval over encrypted data, these schemes only support exact or fuzzy keyword search, mainly evaluate the similarity of keywords from the structure but the semantic relatedness is not considered. This work focuses on realizing secure semantic search through query keyword semantic extension based on the co-occurrence of equivalent terms of query keyword. To achieve efficiency of the search method we enhance the TFIDF algorithm by extending the keyword set with words which are equivalent to keywords. This will ultimately support data retrieval on querying semantic query. Even when user doesn't know exact or synonym of keywords of encrypted data, he can try searching it by its meaning in natural language. Indexed B+ Tree makes the search scheme even more reliable and better.

Keywords: Cloud computing, Encrypted cloud, Privacy Preserving, Multi-keyword search, Semantic based search, E-TFIDF.

ARTICLE INFO

Article History

Received: 22nd June 2016

Received in revised form :

22nd June 2016

Accepted: 25th June 2016

Published online :

29th June 2016

I. INTRODUCTION

Today, consumer centric cloud computing is a new model of enterprise-level in IT infrastructure providing the on-demand high quality applications and services from a shared pool of computing resources. The Cloud Service Provider (CSP) has full control of the outsourced data; it may learn some additional information from that data therefore some problems arise in the circumstance. So, sensitive data is encrypted before outsourcing to the cloud. However the encrypted data make the traditional plaintext search methods useless. The simple and awkward method is downloading all data and decrypt it locally is obviously impractical, because the consumers want to search only the interested data rather all the data. The existing search approaches like ranked search, multi-keyword and semantic based search has been proposed. The Vector Space Model is used to address multi-keyword search and result ranking. By using VSM document index is build and then cosine measure is used to calculate the similarity between the document and the search query. To enhance the efficiency of the search method we use the extended keyword set with words which are most equivalent with query keyword. Even when user doesn't know exact or synonym of keywords of encrypted data, he can try searching it by its meaning in natural language. This makes the Semantic search more efficient and cost can be minimized by employing these scheme into the indexed B+ tree data structure and also we are incorporating hybrid cryptosystem which makes the search scheme even more secured and privacy preserving.

II. PROPOSED SYSTEM

To overcome the problem of effective search system this work proposes an efficient and flexible searchable scheme that supports both privacy preserved and multi-keyword ranked search.

A. Privacy preserved search over encrypted cloud

There are two approaches available for encrypting data i.e. Symmetric encryption and Asymmetric encryption. Both techniques are having their own features and security metrics. So here we are taking advantage of these two techniques by making hybrid cryptosystem which provides us with authentication and confidentiality of sensitive data.
The proposed workflow is:

i. Create a random key for symmetric encryption of user data
ii. Encrypt the data using this random key
iii. Encrypt the random key using asymmetric encryption
iv. Send the encrypted message and the encrypted key to the recipient of search results

B. Multi-keyword Ranked Search

The existing systems like exact or fuzzy keyword search, supports only single keyword search. These schemes doesn't retrieve the relevant data to users query therefore multi-keyword ranked search over encrypted cloud data remains a very challenging problem. To meet this challenge of effective search system, an effective and flexible searchable scheme is proposed that supports multi-keyword ranked search. To address multi-keyword search and result ranking, Vector Space Model (VSM) is used to build document index, that is to say, each document is expressed as a vector where each dimension value is the Term Frequency (TF). The vector has the same dimension with document index and its each dimension value is the Inverse Document Frequency (IDF) weight. Then cosine measure can be used to compute similarity of one document to the search query [1]. To improve search efficiency, a tree-based index structure used which is a balance binary tree is. The searchable index tree is constructed with the document index vectors. So the related documents can be found by traversing the tree.While user searching the data on cloud server it might be possible that the user is unaware of the exact words to search, i.e. there is no tolerance of synonym substitution or syntactic variation which are the typical user searching behaviors and happen very frequently. To solve this problem semantic based search method is used. Here the enhanced E-TFIDF algorithm is proposed for improving documental searches optimized for specific scenarios where user want to find a document but don't remember the exact words used, if plural or singular words were used or if a synonym was used. The defined algorithm takes into consideration:

i. The number of direct words of the search expression that are in the document;
ii. The number of equivalent words related to search query in the search expression that are in the document.[7].

III.MATHEMATICAL MODEL

Using set theory, Efficient Ranked Multi-keyword Search for Encrypted Cloud with Provision of Equivalent Query can be expressed as follows:

Useful Notations:

- DC–plaintext document collection, denoted as a set of n documents DC = $(d_1, d_2, \ldots, d_n)$.
- C–the encrypted document collection for DC, denoted as C = $(C_1, C_2, \ldots, C_n)$.
- W–the keyword dictionary composed of m keywords, denoted as W = $(W_1, W_1, \ldots, W_m)$.
- I–the searchable index tree, which is stored in data owner side.
- eI–the encrypted form of I, which will be outsourced into the cloud server.
- T–the trapdoor generated by the access control mechanism for search request.
- Q–the query vector for the keyword set.
- Du–an index vector stored in the node u of the index tree.
- fDu–the encrypted form of Du.

Let the Proposed system can be expressed as

MRSS = {F, K, I, O, SS} Where

• F - The Plaintext File Collection

• K - Extended Keyword set with semantic Words

• I - Input keywords given by user

• O - Output set containing no. of relevant documents

• SS - Searchable Scheme

The searchable Scheme SS=(KeyGeneration, FileUpload, DataRetrive, Search) is secure.

K = $\{K_{sim1}, K_{sim2}, K_{sim3}, K_{sim4}, \ldots\}$

F = { F1,F2,F3, ...}

IV.SYSTEM ARCHITECTURE

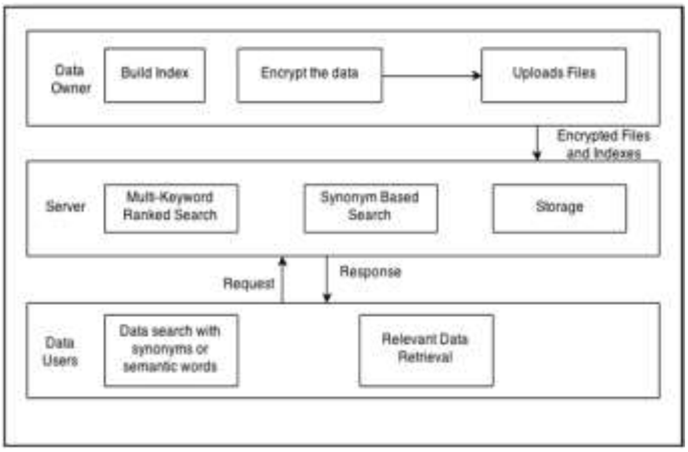


Fig. 1. System Architecture of Efficient Ranked Multi-keyword Search for Encrypted Cloud with Provision of Equivalent Query

Modules description is as follows:

1. Keyword Expansion

In order to improve the accuracy of search results, the keywords are extracted from outsourced text documents need to be extended by common synonyms or equivalent words, as cloud customers' searching input might be the synonyms of the predefined keywords, not the exact or fuzzy matching keywords. due to the lack of exact knowledge about the data.

Then the keyword set is extended by using the constructed synonym thesaurus.

2. Upload Encrypted Data

This module is used to help the data owner to encrypt the document using AES Algorithm and then upload the encrypted document to the cloud for storage purpose. This allows data owner to store their secret key in very secure manner without exposing it to the users of system. For this, secret key is stored again in encrypted form.

3. Search Module

This module helps users to enter their query keyword to get the most relevant documents from set of uploaded documents. This module retrieves the documents from cloud which matches the query keyword.

4. Rank Generation

In information retrieval, a ranking function is usually used to evaluate relevant scores of matching files to a request. The rank function based on the term frequency (TF) and inverse document frequency (IDF) is used in extended format i.e. ETF-IDF. Also this system provides user with most popular documents for their keyword by analysing history of most downloaded documents for particular query keyword.

5. Download Ranked Results

Users can download the resultant set of documents only if he/she is authorized user who has granted permission from data owner to download particular document. Owner will send encrypted secret key and session key to user to decrypt the document.

V. Algorithms Used

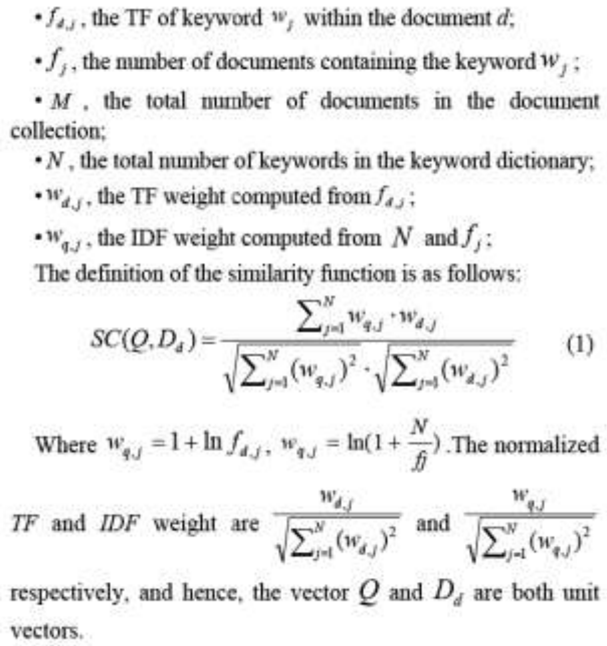
A. E-TFIDF

TF:

TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document).

IDF:

IDF(t) = log_e(Total number of documents / Number of documents with term t in it).



- $f_{d,j}$, the TF of keyword w_j within the document d;
- f_j, the number of documents containing the keyword w_j;
- M, the total number of documents in the document collection;
- N, the total number of keywords in the keyword dictionary;
- $w_{d,j}$, the TF weight computed from $f_{d,j}$;
- $w_{q,j}$, the IDF weight computed from N and f_j;

The definition of the similarity function is as follows:

$$SC(Q,D_d)=\frac{\sum_{j=1}^{N} w_{q,j}\cdot w_{d,j}}{\sqrt{\sum_{j=1}^{N}(w_{q,j})^2}\cdot\sqrt{\sum_{j=1}^{N}(w_{d,j})^2}} \quad (1)$$

Where $w_{q,j}=1+\ln f_{d,j}$, $w_{q,j}=\ln(1+\frac{N}{fj})$. The normalized TF and IDF weight are $\frac{w_{d,j}}{\sqrt{\sum_{j=1}^{N}(w_{d,j})^2}}$ and $\frac{w_{q,j}}{\sqrt{\sum_{j=1}^{N}(w_{q,j})^2}}$ respectively, and hence, the vector Q and D_d are both unit vectors.

B. RSA

1. Key Generation:
   - Pick two large prime numbers p & q, p!=q;
   - Calculate n= p*q;
   - Calculate phi(n)= (p-1)(q-1);
   - Pick e, so that gcd(e,phi(n))=1, 1<e<phi(n);
   - Calculate d, so that d* e mod[phi(n)]=1;
   - Get public Key as (e,n);
   - Get private key as (d,n);
2. Encryption:
   - $C= P^e$ mod n
3. Decryption:
   - $P=C^d$ mod n

C. AES

AES is an iterative rather than Feistel cipher. It is based on ‘substitution–permutation network’. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations). Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix

D. In-Degree Ranking

It is the simple heuristic that can view as a pre-decessor to the link analysis ranking (LAR) ranks the pages according to popularity of the page. The popularity of these pages is calculated by how many pages that are linked to the page. In-degree is measured for ranking any page that is shown in the graph. This simple technique that is heuristic method was applied in many search engines

VI. RESULTS

Table I. File name with Keyword count

Sr. no	File name	Keyword Count
1	Abc.des	22
2	Java.des	10
3	Net.des	15

Table II. File name with Download count

Sr. no	File name	Download Count
1	Abc.des	10
2	Java.des	5
3	Net.des	8

VII. Conclusion

The proposed multi-keyword ranked Search methodology results in the more efficient search process which reduces the network traffic and download bandwidth. The proposed scheme returns the exactly matched files, as well as the files which include the terms semantically relevant to the query keyword. It offers appropriate semantic distance between terms to accomplish the query keyword extension. The encryption has been implemented to guarantee the security and efficiency of data, before it is outsourced to cloud, and provides protection to datasets, indexes and keywords.

REFRENCES

.

[1] Zhangjie Fu, Xingming Sun, Nigel Linge and Lu Zhou, "Achieving Effective Cloud Search Services: Multi-keyword Ranked Search over Encrypted Cloud Data Supporting Synonym Query", IEEE Transactions on Consumer Electronics, Vol. 60, No. 1, February 2014.

[2] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy keyword search over encrypted data in cloud computing", Proceedings of IEEE INFOCOM'10 Mini-Conference, San Diego, CA, USA, pp. 1-5, Mar. 2010.

[3] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure ranked keyword search over encrypted cloud data", Proceedings of IEEE 30th International Conference on Distributed Computing Systems (ICDCS), pp. 253-262, 2010.

[4] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data", IEEE Transactions on Parallel and Distributed Cloud Computing Systems,Volume:25,Issue:1,Issue Date:Jan.2014.

[5] Q. Chai, and G. Gong, "Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers", Proceedings of IEEE International Conference on Communications (ICC'12), pp. 917-922, 2012.

[6] W. Sun, B. Wang, N. Cao, M. Li, W. Lou, and Y. T. Hou, "Privacy preserving multi-keyword text search in the cloud supporting similarity based ranking", ASIA CCS 2013, Hangzhou, China, May 2013, ACM pp. 71-82, 2013.

[7] Sara Paiva,"A Fuzzy Algorithm for Optimizing Semantic Documental Searches", International Conference on Project Management / HCIST 2013.

[8] Tyne Liang and Dian-Song Wu, "Automatic Pronominal Anaphora Resolution in English Texts".