Journal of
Information Technology & Software Engineering

Kaur and Mahajan, J Inform Tech Softw Eng 2015, 5:2
DOI: 10.4172/2165-7866.1000152

Research Article Open Access

An Approach towards Efficient Ranked Search over Encrypted Cloud

Rajpreet Kaur[1]*and Manish Mahajan[2]

[1]CGC, Landran Mohali, Punjab, India
[2]Computer Science and Engineering, CGC, Landran Mohali, Punjab, India

Abstract

In present, Cloud computing is a dominant field in information technology. With the increased rate of data outsourcing over cloud data privacy of sensitive data becomes a big issue. For the security purpose data is encrypted before outsourcing. But encrypted data is very difficult to be retrieved efficiently. Although some traditional search schemes are available for searching encrypted data, but these techniques are merely base on Boolean search and not deals with the relevance of files. These approaches suffer from two main shortcomings. Firstly, if one user has no pre-knowledge of encrypted data, has to process every retrieved file to find results of his use. Secondly, every time retrieving all the files containing query keyword increases network traffic. This work is dedicated to develop an approach for secure and effective retrieval of cloud data. Ranked search greatly improves the performance by returning the files in ranked order based on some similarity relevance criteria. To achieve more practical performance, system demonstrates an approach for symmetric searchable encryption (SSE) which utilizes information retrieval and cryptography primitives. Hence the implementation is based on order-preserving symmetric encryption (OPSE).

Keywords: Cloud; Data privacy; Similarity relevance; Ranking; SSE

Introduction

Cloud computing can be assumed as a model for delivering information technology services (like storage space, networking, applications etc) in which resources are retrieved from internet using web based tools, rather than a direct connection to server. Cloud computing provides hardware and software resources from a shared pool of resources on rent according to user's demand. So this technology releases user from burdens of management efforts and also from headaches of installation and maintenance.

Service model

Cloud software as a service (SaaS): In this software is made available to the user as service. Cloud applications are generally accessible from various devices like mobile, tablet, laptop, PC, workstations, servers etc. The user has no control over the underlying platform and infrastructure. Examples are Dropbox, Gmail, Gtalk etc.

Cloud platform as a service (PaaS): In this software is made available to the user as service. Programming languages and tools are provided by service provider to develop and deployment services. A user has no control over underlying infrastructure but has control over the deployed applications. Examples are Windows Azure, Google App Engine.

Cloud infrastructure as a service (IaaS): In this Software is made available to the user as a service. A user can demand computing infrastructure, storage infrastructure and network infrastructure etc from service provider. User is not the actual owner of the infrastructure but has control over operating systems, storage, deployed applications etc. For example Amazon, Rack space cloud.

Searching Cloud Data

As cloud computing has become a prevalent platform in information technology, the amount of sensitive information centralized over cloud is also increasing. These information files contain confidential data like personal medical records, government documents, private photos etc. To protect privacy of data and to prevent unauthorized access, it becomes very necessary to encrypt data before outsourcing to ensure data integrity and confidentiality. Along with this, data owner may share their outsourced data with a number of users. But, each user desires to retrieve files of his own interest during a given time period, which makes data utilization very challenging. From the existing approaches, most common is using keyword based search technique. These techniques are commonly applied for plaintext search scenarios where user can retrieve the files of interest by giving keyword in query. Unluckily, data encryption for securing outsourced data makes these traditional methods to become fail for searching cloud data. Although, some traditional encryption techniques facilitate user to search over encrypted data without first decrypting it. But these techniques only support Boolean search, where files are retrieved according to presence or absence of keyword in file and don't consider relevance of files (Figure 1).

Many existing approaches for ranked order search and relevance score of files are being used by Information retrieval (IR) community for searching cloud data. Although the importance of ranked search is

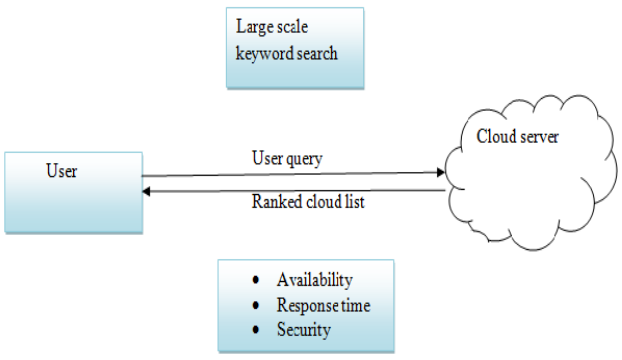

Figure 1: Architecture of keyword search.

***Corresponding author:** Rajpreet Kaur, Research Scholar, CGC, Landran Mohali, Punjab, India, Tel: 0172 398 4200; Email: Preet.billing00@gmail.com

Received June 17, 2015; **Accepted** July 10, 2015; **Published** July 20, 2015

Citation: Kaur R, Mahajan M (2015) An Approach towards Efficient Ranked Search over Encrypted Cloud. J Inform Tech Softw Eng 5: 152. doi:10.4172/2165-7866.1000152

receiving attention from a long time, yet the topic of encrypted search is not addressed much. Therefore enabling a mechanism for secure symmetric encryption and ranked search is a problem tackled in this work.

Related Work

Early searchable encryption techniques were only base on exact keyword search [1-6]. Song et al. gave an SE scheme for symmetric keyword search. In which each file in the file is encrypted using two-layered encryption method [1]. Index creation is used by some researchers for efficiency improvement. In index based techniques secure index is constructed for every unique word in a file [2,6]. In the work proposed by Curtmola et al., for each keyword index construction, entries are done to hash table. Each entry contains index for unique word and their trapdoor file identifier [5].

Further, some researchers stepped towards ranked search to improve usability. Wang et al. [7,8] proposed ranked search mechanism based on certain relevance scores to identify similarity of files with queried keyword. This approach was single-keyword based. Moving a step ahead, multi-keyword search is elaborated by Yang et al. and Cao et al. [9,10]. They used "similarity based inner product for result ranking.

However, all above schemes are exact keyword search based. For enhancing search flexibility, fuzzy- keyword based search is introduced by some authors [11-13]. Edit distance concept is used for calculating similarity of keywords with each other for generating fuzzy keyword sets for indexes. Li et al. and Wang et al. [11,14] presented this edit distance procedure. In this paper an implementation is given for secure symmetric search encryption and ranking of results in a particular order according to some relevance criteria.

Searchable Symmetric Encryption

System Model- Basic model involves three types of entities which are represented in the Fig. These entities are named as Data owner (O), cloud server (CS) and user (U). A collection of n data files $C=(F_1, F_2, \ldots, F_n)$ is outsourced by data owner onto the cloud into encrypted format. However for effective utilization of encrypted files data owner creates secure index I using a set m of different keywords given by $W=(w_1, w_2, \ldots, w_m)$, which are extracted from file collection C. After this both encrypted data and indexes are outsourced onto the cloud server.

When an authorized user wants to retrieve some data file which contains some keyword w, a secret search trapdoor is created by user to the cloud server.

When server receives the search request T_w, all the computation is done by server. After checking the index I, server returns the matching files to user. Project considers the secure ranked keyword search problem as follows: the search result should be returned according to certain ranked relevance criteria (e.g., keyword frequency), to improve file retrieval accuracy for users without prior knowledge on the file collection C. w (Figure 2).

All the computation work for finding search results and generating relevance score of files is done at cloud server. Hence the server has the access to all the files which are stored. For security purpose server should act in an "honest" manner and strictly follows the assigned protocols.

It considers an "honest-but-curious" server in our model. In other words, the cloud server has no intention to actively modify the message flow or disrupt any other kind of services.

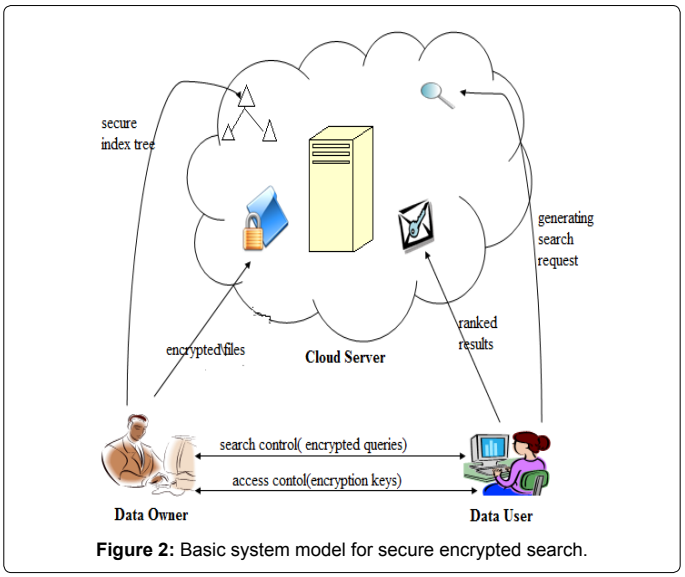

Figure 2: Basic system model for secure encrypted search.

Index and semantic relationship library (SRL) for various unique words is constructed using metadata created. When user gives some query keyword, server expands it on the basis of SRL. After searching index, it returns the relevant files to user.

Design goals

To enable ranked searchable symmetric encryption for effective utilization of outsourced cloud data under the aforementioned model, project design should achieve the following security and performance guarantee. Specifically, it has the following goals:

- Ranked keyword search: To explore various existing mechanisms for secure searchable encryption and to build a framework for effective ranked search.
- Security guarantee: To making the outsourced data secure by preventing cloud server from learning plaintext of files.
- Achieving Efficiency: Above goals should be achieved with minimum communication and computation overhead.

Notation and preliminaries

- C – the file collection to be outsourced, denoted as a set of n data files $C=(F_1, F_2, \ldots, F_n)$.
- W – the distinct keywords extracted from file collection C, denoted as a set of m words $W=(w_1, w_2, \ldots, w_m)$.
- id(F_j) – the identifier of file F_j that can help uniquely locate the actual file.
- I – the index built from the file collection, including a set of posting lists $\{I(w_i)\}$, as introduced below.
- T_{wi} – the trapdoor generated by a user as a search request of keyword w_i.
- $F(w_i)$ – the set of identifiers of files in C that contain keyword w_i.
- N_i – the number of files containing the keyword w_i and
- $N_i=|F(w_i)|$.

Further project introduces some necessary information retrieval background for our proposed system:

Inverted index

Inverted index (also referred as posting files) is widely used indexing scheme in information retrieval. In inverted index structure a unique index value is given to every keyword and list of mappings is generated from keywords to the files in which word is present. For enabling ranked search, a relevance score for files is calculated using some mathematical assumptions.

Ranking function

A ranking function is used to compute similarity of terms by calculating relevance score. For a given search request, score is generated for matching files which are relevant to queried keyword. The most widely used statistical measurement for evaluating relevance score in the information retrieval community uses the TF × IDF rule, where TF (term frequency) is simply the number of times a given term or keyword appears within a file (to measure the importance of the term within the particular file), and IDF (inverse document frequency) is obtained by dividing the number of files in the whole collection by the number of files containing the term (to measure the overall importance of the term within the whole collection).

Order preserving symmetric encryption

The OPSE is a deterministic encryption scheme where the numerical ordering of the plaintexts gets preserved by the encryption function. Boldyreva et al. [15] gives the first cryptographic study of OPSE primitive and implements a secure search framework using pseudorandom function and permutation. This work considers an order-preserving function $g(\cdot)$ from domain $D=\{1, \ldots, M\}$ to range $R=\{1, \ldots, N\}$, which can be uniquely defined by a combination of M out of N ordered items. An OPSE can be said secure only if an attacker has to perform a brute force search over all the possible combinations of M out of N to break the encryption scheme. If the security level chosen is of 64 bits, then it is good to choose $M=N/2>64$, in order to create number of combinations so that the total number of combinations will be greater than 264. This construction is based on relationship between order preserving function and hyper geometric probability distribution (HGD). Their construction is based on an uncovered relationship between a random order-preserving function (which meets the above security notion) and the hyper geometric probability distribution, which will later be denoted as HGD. Readers can refer [15] for more details of OPSE. As first look, It seems changing relevance based encryption from earlier search schemes to OPSE is very efficient. But OPSE is deterministic encryption scheme, in which if data is not handled appropriately, then a little mistake can leaks lots of information.

Problem Statement

Problem formulation

In the early techniques for symmetric search like fuzzy keyword search etc, were mainly used for searching. However, these techniques enhance search flexibility and usability. They consider structure of terms and edit distance between terms to calculate similarity [16]. But don't consider the terms semantically related to search keyword. The results were only based on presence or absence of keyword. For example these schemes only consider certain misspelling or inconsistencies like "Written" or "written" are considered to be similar. The most important thing which was Result-ranking was still out of ordering.

Implementation of system

Semantic expansion based similar search enhances usability by returning exactly matching files and also returns the files which are relevant to given query keyword. From the metadata set cloud server generates the inverted index and constructs the semantic relationship library (SRL) for keywords set. Cloud server automatically finds all relevance files using SRL, when user makes a search request.

In the implemented system, to ensure security and final result ranking, order-preserving encryption is used to preserve numerical ordering for protecting relevance score.

The above straightforward approach demonstrates the core problem that causes the inefficiency of ranked searchable encryption. Server should perform searching and ranking quickly by not knowing relevance score and other information of files.

The main objectives of the given scheme are discussed below:

- To design a search scheme for encrypted cloud data that gives relevance score to files with query keyword and returns the retrieved files in order.
- To enable efficient utilization of data files using ranked searchable encryption scheme.
- To enable security by preventing cloud server fron learning plaintext of data files.

Methodology

Implementation of discussed work is shown by the flowchart in the figure given below (Figures 3 and 4).

Steps of implementation

1. In first step, encryption of data is done using AES (Asymmetric encryption standard) algorithm. The implementation is done by generating local environment in MATLAB. Different GUI (Graphical user interface) are created for user interaction. This algorithm encrypts the data file and also creates index value for every unique keyword [17] (Figure 5).

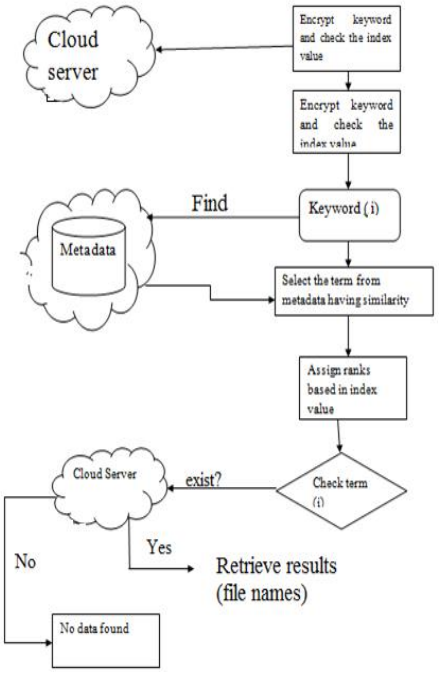

Figure 3: Flowchart of work.

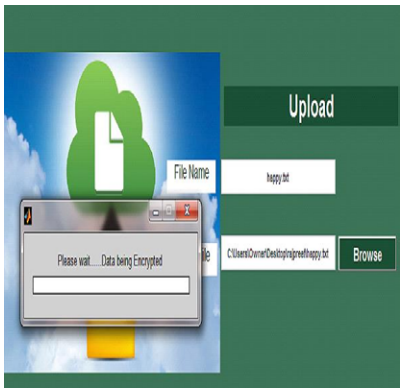

Figure 4: Encryption process of uploaded data.

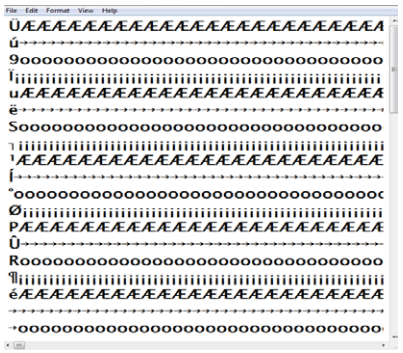

Figure 5: Encrypted file.

File ID	Relevance Score
F1	6.52
F2	3.42
F3	2.29

Figure 6: Table for relevance score.

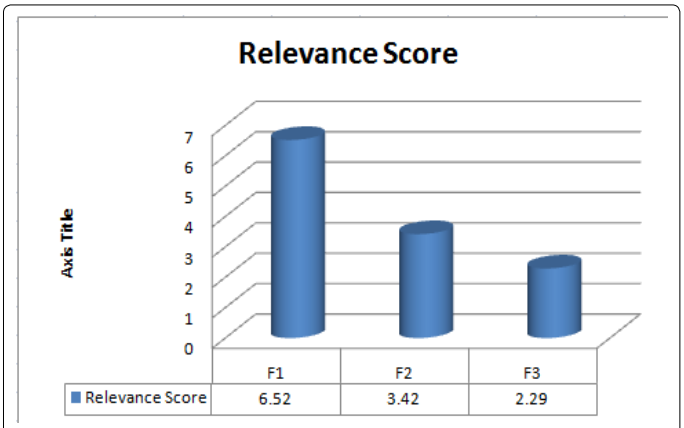

Figure 7: Graphical representation of relevance scores.

2. After encryption and indexing Metadata is created for keywords. In metadata relative terms are represented.

3. Now if a user searches a term by giving query keyword, then SSE (Secure symmetric encryption) and OPSE are used for giving results in ranked order.

Result Analysis

The experimental results can be explained with the set of some snapshots.

1. First of all a data owner uploads some data file, which is encrypted using AES algorithm.

2. The uploaded on cloud server can be searched in a secure way using SSE (secure symmetric encryption) algorithm. The search results are displayed in ranked form using OPSE (order preserving symmetric encryption

3. Algorithm generate the relevance score of files based on term frequency (TF) and inverse domain frequency (IDF), using the equation TF×IDF. Where TF can be defined as the number of times given keyword or term exists in a given file. IDF can be calculated by dividing the number of files in whole collection by number of files containing that keyword (Figures 6 and 7).

4. Hence based on relevance score files can be ranked for more symmetry.

References

1. Song DX, Wagner D, Perrig A (2000) Practical techniques for searches on encrypted data. Proceedings of IEEE Symposium on Security and Privacy, IEEE, Berkeley, California.
2. Goh EJ (2003) Secure indexes. Cryptology ePrint Archive, Report 2003/216
3. Boneh D, Cresecenzo G, Ostrovsky R, Persiano G (2004) Public key encryption with keyword search. Advances in Cryptology- Eurocrypt 3027: 506-522.
4. Chang YC, Mitzenmacher M (2005) Privacy preserving keyword searches on remote encrypted data. Applied Cryptography and Network Security 3531: 442-455.
5. Curtmola R, Garay J, Kamara S, Ostrovsky R (2006) Searchable symmetric encryption: improved definitions and efficient constructions. Proceedings of the 13th ACM conference on Computer and communications security, ACM, Alexandria, VA, USA.
6. Bellare M, Boldyreva A, O'Neill A (2007) Deterministic and efficiently searchable encryption. In Advances in Cryptology-CRYPTO 4622: 535-552.
7. Wang C, Cao N, Li J, Ren K, Lou W (2010) Secure ranked keyword search over encrypted cloud data. 30th IEEE International Conference on Distributed Computing Systems (ICDCS), IEEE Comp Society Washington, DC, USA.
8. Wang C, Cao N, Ren K, Lou W (2012) Enabling secure and efficient ranked keyword search over outsourced cloud data. IEEE Trans Parallel Distrib Syst 23:1467-1479.
9. Cao N, Wang C, Li M, Ren K, Lou W (2011) Privacy-preserving multi-keyword ranked search over encrypted cloud data. Proceedings of IEEE INFOCOM. IEEE, Shanghai.
10. Yang C, Zhang W, Xu J, Xu J, Yu N (2012) A Fast Privacy-Preserving Multi-keyword Search Scheme on Cloud Data. International Conference on Cloud and Service Computing (CSC). IEEE, Shanghai, China.
11. Wang C, Ren K, Yu S, Urs, KMR (2012) Achieving Usable And Privacy-assured similarity search over outsourced cloud data. Proceeding of IEEE INFOCOM, Orlando, Florida, USA.
12. Li J, Wang Q, Wang C, Cao N, Ren K, et al. (2010) Fuzzy keyword search over encrypted data in cloud computing. Proceedings of IEEE INFOCOM. IEEE, San Diego, CA, USA.
13. Xia Z, Zhu Y, Sun X, Chen L (2014) Secure semantic expansion based search over encrypted cloud data supporting similarity ranking. Journal of Cloud Computing 3.
14. Stefanov E, Papamanthou C, Shi E (2014) Practical Dynamic Searchable Encryption with Small Leakage. NDSS '14, San Diego, CA, USA, pp. 23-26.

15. Boldreva A, Chenette N, Lee Y, O'neill A (2009) Order-preserving Symmetric encryption. Advances in Cryptology EUROCRYPT 2009 Springer, Berlin, Heidelberg 5479: 224-241.

16. Bellare M, Boldyreva A, O'Neill A (2007) Deterministic and efficiently searchable encryption. Advances in Cryptology CRYPTO, Springer, Berlin,Heidelberg 4622: 535- 552.

17. Chuah M, Hu W (2011) Privacy-aware bed tree based solution for fuzzy multi-keyword search over encrypted data. 31st International Conference on Distributed Computing Systems Workshops (ICDCSW). IEEE, Minneapolis, Minnesota, USA.