SEMANTIC COMPOUND KEYWORD SEARCH OVER ENCRYPTED DATA IN CLOUD COMPUTING

[1]Vegi Srinivas, [2]Y. Dinesh Kumar, [3]K. Sujatha
[1]Associate Professor, [2]Assistant Professor, [3]Professor
[1]Department of Computer Science and Engineering
[1]Dadi Institute of Engineering & Technology, Anakapalle, India.

***Abstract*:** Cloud storage becomes more and more popular in the recent trend since it provides various benefits over the traditional storage solutions. Along with many benefits provided by cloud storage, many security problems arise in cloud storage which prevents enterprises from migrate their data to cloud storage. These security problems induce the data owners to encrypt all their sensitive data. The encryption approach may have strengthened the data security of cloud data, but it degrades the data efficiency because the encryption reduces the search ability of the data. Many schemes were proposed in recent researches which enable keyword search over encrypted data in cloud computing, and these schemes contain weaknesses which make them impractical when applying these schemes in real life scenarios. Hence, semantic-based keyword search over encrypted cloud data becomes of paramount importance. In this paper, we propose a semantic-based compound keyword search (SCKS) scheme is proposed. SCKS achieves not only semantic-based search but also multi-keyword search and ranked keyword search. Additionally, SCKS also eliminates the predefined global library and can efficiently support data update. The experimental results on real-world dataset indicate that SCKS introduces low overhead on computation and the search accuracy outperforms the existing schemes.

Index Terms: Cloud Storage, Encryption, Keyword Search, Symmetric Searchable Encryption.

1.Introduction

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers [1].

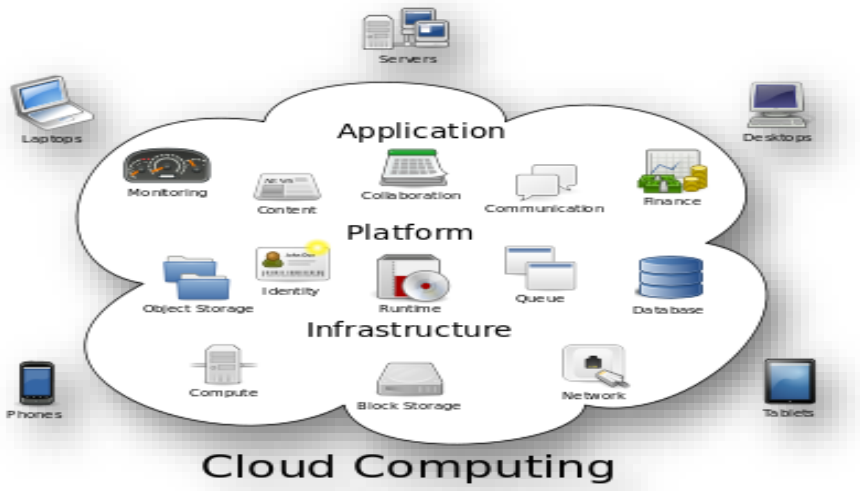

Fig 1. Cloud Computing Architecture

1.1 How Cloud Computing Works?

The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive computer games.

The cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing.

1.2 Characteristics and Services Models:

The salient characteristics of cloud computing based on the definitions provided by the National Institute of Standards and Terminology (NIST) are outlined below:

- **On-demand self-service**: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.

- **Broad network access**: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).
- **Resource pooling**: The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.
- **Rapid elasticity**: Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.
- **Measured service**: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be managed, controlled, and reported providing transparency for both the provider and consumer of the utilized service [2].

2. Literature Review

K. Ren, C.Wang, Q.Wang et al. in [3] said that cloud computing represents today's most exciting computing paradigm shift in information technology. However, security and privacy are perceived as primary obstacles to its wide adoption. Here, the authors' outline several critical securities challenges and motivate further investigation of security solutions for a trustworthy public cloud environment.

C. Gentry et al. in [4] discussed fully homomorphic encryption scheme, solving an old open problem. Such a scheme allows one to compute arbitrary functions over encrypted data without the decryption key—i.e., given encryptions E(m1), ..., E(mt) of m1, ..., m t, one can efficiently compute a compact cipher text that encrypts f(m1, ..., m t) for any efficiently computable function f. Fully homomorphic encryption has numerous applications. For example, it enables encrypted search engine queries—i.e., a search engine can give you a succinct encrypted answer to your (boolean) query without even knowing what your query was. It also enables searching on encrypted data.

D. Boneh, G. Di Crescenzo et al. in [5] studied the problem of searching on data that is encrypted using a public key system. Consider user Bob who sends email to user Alice encrypted under

Alice's public key. An email gateway wants to test whether the email contains the keyword "urgent" so that it could route the email accordingly. Alice, on the other hand does not wish to give the gateway the ability to decrypt all her messages. They define the concept of public key encryption with keyword search and give several constructions.

D. X. Song, D. Wagner, and A. Perrig[6], have proposed that the store data on data storage servers such as mail servers and file servers in encrypted form to reduce security and privacy risks. But this usually implies that one has to sacrifice functionality for security. For example, if a client wishes to retrieve only documents containing certain words, it was not previously known how to let the data storage server perform the search and answer the query, without loss of data confidentiality. They describe cryptographic schemes for the problem of searching on encrypted data and provide proofs of security for the resulting crypto systems. The algorithms presented are simple, fast (for a document of length n, the encryption and search algorithms only need O(n) stream cipher and block cipher operations), and introduce almost no space and communication overhead, and hence are practical to use today .

C. Chang and M. Mitzenmacher in [7] consider the following problem: a user U wants to store his files in an encrypted form on a remote file server S. Later the user U wants to efficiently retrieve some of the encrypted files containing (or indexed by) specific keywords, keeping the keywords themselves secret and not jeopardizing the security of the remotely stored files. For example, a user may want to store old e-mail messages encrypted on a server managed by Yahoo or another large vendor, and later retrieve certain messages while travelling with a mobile device.

In this paper, we offer solutions for this problem under well-defined security requirements. Our schemes are efficient in the sense that no public-key cryptosystem is involved. Indeed, our approach is independent of the encryption method chosen for the remote files. They are also incremental, in that U can submit new files which are secure against previous queries but still searchable against future queries.

3. System Design

3.1 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required,

controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

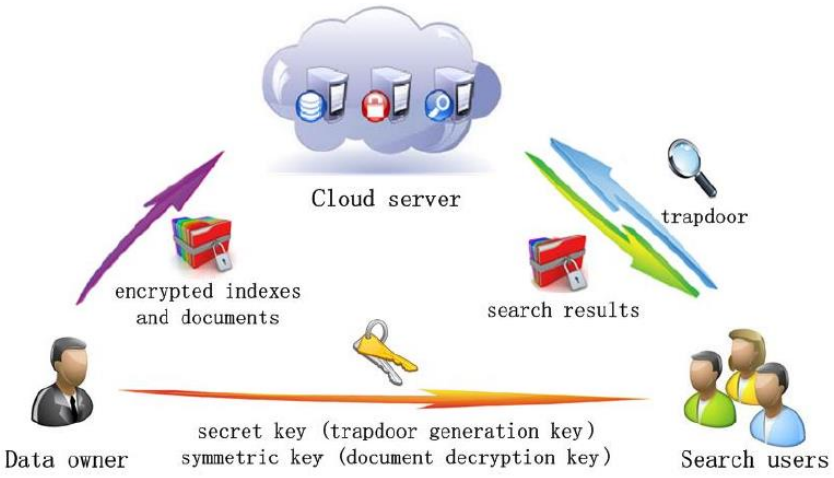

Fig 2. Proposed model for keyword search over encrypted data

3.2 OBJECTIVES

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

3.3 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for

immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

4. Experimental Results

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 Searchable Encryption (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 [8]. 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.

Figure 3 present overall graphical representation of mean precision ratio of search engine for first 20 documents.

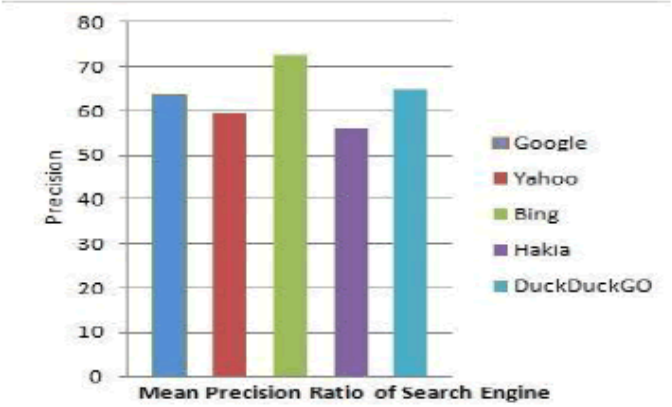

Fig. 3. Precision ratio of search engines for first 20 documents

5. Conclusion and Future Enhancements

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.

6. References

[1] R. L Grossman, "The Case for Cloud Computing", IT Professional, vol. 11(2), pp. 23-27, 2009, ISSN: 1520-9202.

[2] Imran Ashraf, "An Overview of Service Models of Cloud Computing", International Journal of Multidisciplinary and Current Research, Vol.2 (July/Aug 2014 issue)

[3] K. Ren, C.Wang, and Q.Wang "Security Challenges for the Public Cloud", View from the Cloud, IEEE Computer Society, January /February 2012.

[4] Craig Gentry Shai Halevi, "Implementing Gentry's Fully-Homomorphic Encryption Scheme", IBM Research, Feb. 2011.

[5] D. Boneh, G. Di Crescenzo, "Public Key Encryption with Keyword Search", Advances in Cryptology – EUROCRYPT 2004, pp. 506-522.

[6] D. X. Song, D. Wagner, and A. Perrig , "Practical techniques for searches on encrypted data". Proceedings of the IEEE Symposium on Security and Privacy, 2000, pp.44-55.

[7] C. Chang and M. Mitzenmacher, "Privacy Preserving Keyword Searches on Remote Encrypted Data", Applied Cryptography and Network Security(ACNS), 2005, pp.442-455.

[8] Xiuxiu Jiang, Xinrui Ge , Jia Yu, Fanyu Kong, "An Efficient Symmetric Searchable Encryption Scheme for Cloud Storage", Journal of Internet Services and Information Security (JISIS), volume: 7, number: 2 (May 2017), pp. 1-18.