IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 2, February 2015.
www.ijiset.com

ISSN 2348 – 7968

A Novel Approach For Multi-Keyword Search In Cloud Computing Using Homomorphic Token Pre-Computation

Pandi Priya.U[1], Padma Priya.R[2]

[1]Department of Computer Science and Information Technology
Nadar Saraswathi College of Arts and Science, Theni dt-625531, Tamilnadu, India

[2]Department of Computer Application
Nadar Saraswathi College of Arts and Science, Theni dt-625531, Tamilnadu, India

Abstract

We propose to investigate the problem of multiple keyword queries for keyword search over a large number of structured and Linked Data sources. Routing keywords only to relevant sources can reduce the high cost of searching for structured results that span multiple sources Existing work uses keyword relationships collected individually for single databases. To the best of our knowledge, the work presented in this paper represents the first attempt to address this problem. We represent relationships between keywords as well as those between data elements. They are constructed for the entire collection of linked sources, and then grouped as elements of a compact summary called the set-level keyword-element relationship. Summarizing relationships is essential for addressing the scalability requirement of the Linked Data web scenario. We design search schemes which allow multi-keyword query and provide result similarity ranking for effective data retrieval, instead of returning undifferentiated results. It prevents the cloud server from learning additional information from the dataset and the index, and to meet privacy requirements. Among Various semantics, we choose the effective principle of "coordinate matching" and KNN Technique to find related words to improve the searching technique. Also uses Homomorphic token pre-computation to provide privacy and achieves with low communication and computation overhead. We also introduced double level security to search and access the data which are stored in cloud.

Keyword: *Multikeywordsearch, co-ordinate matching, Homomorphic token Pre-computation.*

1. Introduction

Cloud Computing

The essence of cloud computing is you can have access to your data through the internet anytime from anywhere in the world. If you have your information stored in your laptop and it crashes, you will have a hard time putting together the information all over again. However, with all your personal information stored in cloud, you are safe from data loss.

However, most businesses have initial concerns about the security that cloud computing offers. Hence, all the CSP (cloud service providers) guarantees you security of your data through the installation of anti-virus and firewalls in your server as well as constantly updates them as and when any new patches are released.

ISSN 2348 – 7968

Cloud computing refers to the delivery of computing resources over the Internet. Instead of keeping data on your own hard drive or updating applications for your needs, you use a service over the Internet, at another location, to store your information or use its applications. Doing so may give rise to certain privacy implications Cloud computing providing unlimited infrastructure to store and execute customer data and program. As customers you do not need to own the infrastructure, they are merely accessing or renting; they can forego capital expenditure and consume resources as a service, paying instead for what they use.

2. Security a Major Concern

- Security concerns arising because both customer data and program are residing in Provider Premises.
- Security is always a major concern in Open System Architectures

Data centre Security?

- Professional Security staff utilizing video surveillance, state of the art intrusion detection systems, and other electronic means.
- When an employee no longer has a business need to access datacenter his privileges to access datacenter should be immediately revoked.
- All physical and electronic access to data centers by employees should be logged and audited routinely.
- Audit tools so that users can easily determine how their data is stored, protected, used, and verify policy enforcement.

Backups of Data

- Data store in database of provider should be redundantly store in multiple physical locations.
- Data that is generated during running of program on instances is all customer data and therefore provider should not perform backups.
- Control of Administrator on Databases.

Data Sanitization

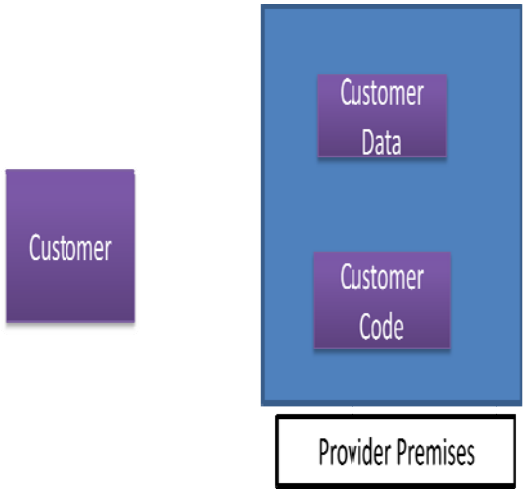

Fig 1 -Open system architecture

- Sanitization is the process of removing sensitive information from a storage device.
- What happens to data stored in a cloud computing environment once it has passed its user's "use by date"
- What data sanitization practices does the cloud computing service provider propose to implement for redundant and retiring

ISSN 2348 – 7968

data storage devices as and when these devices are retired or taken out of service.

Network Security

- **Denial of Service**: where servers and networks are brought down by a huge amount of network traffic and users are denied the access to a certain Internet based service.
- **Like DNS Hacking**: Routing Table "Poisoning", XDoS attacks
- **QoS Violation**: through congestion, delaying or dropping packets, or through resource hacking.
- **Man in the Middle Attack**: To overcome it always use SSL
- **IP Spoofing**: Spoofing is the creation of TCP/IP packets using somebody else's IP address.
- **Solution**: Infrastructure will not permit an instance to send traffic with a source IP or MAC address other than its own.

How secure is encryption Scheme

- Is it possible for all of my data to be fully encrypted?
- What algorithms are used?
- Who holds, maintains and issues the keys? Problem:
- Encryption accidents can make data totally unusable.
- Encryption can complicate availability Solution
- The cloud provider should provide evidence that encryption schemes were designed and tested by experienced specialists.

Information Security

- Security related to the information exchanged between different hosts or between hosts and users.
- This issues pertaining to ***secure communication***, ***authentication***, *and issues* concerning ***single sign on*** *and* ***delegation***.
- Secure communication issues include those security concerns that arise during the communication between two entities.
- These include confidentiality and integrity issues. Confidentiality indicates that all data sent by users should be accessible to only "legitimate" receivers, and integrity indicates that all data received should only be sent/modified by "legitimate" senders.
- **Solution:** public key encryption, X.509 certificates, and the Secure Sockets Layer (SSL) enables secure authentication and communication over computer networks.

Cloud computing is the long dreamed vision of computing as a utility, where cloud customers can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. Its great flexibility and economic savings are motivating both individuals and enterprises to outsource their local complex data management system into the cloud. To protect data privacy and combat unsolicited accesses in the cloud and beyond, sensitive data, e.g., emails, personal health records, photo albums, tax documents, financial transactions, etc., may have to be encrypted by data owners before outsourcing to the commercial public cloud; this, however, obsoletes the traditional data utilization service

based on plaintext keyword search. The trivial solution of downloading all the data and decrypting locally is clearly impractical, due to the huge amount of bandwidth cost in cloud scale systems. Moreover, aside from eliminating the local storage management, storing data into the cloud serves no purpose unless they can be easily searched and utilized. Thus, exploring privacy-preserving and effective search service over encrypted cloud data is of paramount importance. Considering the potentially large number of on-demand data users and huge amount of outsourced data documents in the cloud, this problem is particularly challenging as it is extremely difficult to meet also the requirements of performance, system usability and scalability.

Our contributions are summarized as follows,

1) For the first time, we explore the problem of multikeyword ranked search over encrypted cloud data, and establish a set of strict privacy requirements for such a secure cloud data utilization system.

2) We propose two MRSE schemes based on the similarity measure of "coordinate matching" while meeting different privacy requirements in two different threat models.

3) Thorough analysis investigating privacy and efficiency guarantees of the proposed schemes is given, and experiments on the real-world dataset further show the proposed schemes indeed introduce low overhead on computation and communication.

3. Description of MRSE Framework

MRSE Framework for easy presentation, operations on the data documents are not shown in the framework since the data owner could easily employ the traditional symmetric key cryptography to encrypt and then outsource data. With focus on the index and query, the MRSE system consists of four algorithms as follows.

- **Setup(1^{ℓ})** Taking a security parameter ℓ as input, the data owner outputs a symmetric key as SK.
- **BuildIndex(F , SK)** Based on the dataset F , the data owner builds a searchable index I which is encrypted by the symmetric key SK and then outsourced to the cloud server. After the index construction, the document collection can be independently encrypted and outsourced.
- **Trapdoor ($\widetilde{W}$)** With t keywords of interest in $\widetilde{W}$ as input ,this algorithm generates a corresponding trapdoor T $\widetilde{W}$.
- **Query(T_W, k, I)** When the cloud server receives a query request as (T_W , k), it performs the ranked search on the index I with the help of trapdoor T_W , and finally returns F_W, the ranked id list of top-k documents sorted by their similarity with $\widetilde{W}$.

Neither the search control nor the access control is within the scope of this paper. While the former is to regulate how authorized users acquire trapdoors, the later is to manage users' access to outsourced documents.

Privacy Requirements for MRSE

The representative privacy guarantee in the related literature, such as searchable encryption, is that the server should learn nothing but search results. With this general privacy description, we explore and establish a set of strict privacy requirements specifically for the MRSE framework.

ISSN 2348 – 7968

Keyword Privacy: As users usually prefer to keep their search from being exposed to others like the cloud server, the most important concern is to hide what they are searching, i.e., the keywords indicated by the corresponding trapdoor. Although the trapdoor can be generated in a cryptographic way to protect the query keywords, the cloud server could do some statistical

Analysis over the search result to make an estimate. As a kind of statistical information, document frequency (i.e., the number of documents containing the keyword) is sufficient to identify the keyword with high probability. When the cloud server knows some background information of the dataset, this keyword specific information may be utilized to reverse-engineer the keyword.

Trapdoor Unlinkability: The trapdoor generation function should be a randomized one instead of being deterministic. In particular, the cloud server should not be able to deduce

The relationship of any given trapdoors, e.g., to determine whether the two trapdoors are formed by the same search request. Otherwise, the deterministic trapdoor generation would give the cloud server advantage to accumulate frequencies of different search requests regarding different keyword(s), which may further violate the aforementioned keyword privacy requirement. So the fundamental protection for trapdoor unlinkability is to introduce sufficient no determinacy into the trapdoor generation procedure.

4. Proposed Work

Homo Morphic Token Pre-computation

Homomorphic encryption is a form of encryption that allows computations to be carried out on ciphertext, thus generating an encrypted result which, when decrypted, matches the result of operations performed on the plaintext. This is sometimes a desirable feature in modern communication system architectures. Homomorphic encryption would allow the chaining together of different services without exposing the data to each of those services

Homomorphic encryptions allow complex mathematical operations to be performed on encrypted data without compromising the encryption.

ALGORITHM

1: **procedure**

2: Choose parameters l, n and function f;

3: Choose the number t of tokens;

4: Choose the number r of indices per verification;

5: Generate master key

6: **for** vector G(j), j ← 1, n **do**

7: **for** round i← 1, t **do**

8: Derive _i = fkchal (i) and k(i) prp from KPRP .

9: Compute v (j)

i =Pr

q=1 * G(j)[_k(i)prp (q)]

10: **end for**

11: **end for**

12: Store all the vis locally.

13: **end procedure**

ISSN 2348 – 7968

In order to achieve assurance of data storage correctness and data error localization simultaneously, our scheme entirely relies on the pre-computed verification tokens. The main idea is as follows: before file distribution the user pre-computes a certain number of short verification tokens on individual vector.

Each token covering a random subset of data blocks. Later, when the user wants to make sure the storage correctness for the data in the cloud, he challenges the cloud servers with a set of randomly generated block indices. Upon receiving challenge, each cloud server computes a short "signature" over the specified blocks and returns them to the user. The values of these signatures should match the corresponding tokens pre-computed by the user. Meanwhile, as all servers operate over the same subset of the indices, the requested response values for integrity check must also be a valid codeword determined by secret matrix P.

Suppose the user wants to challenge the cloud servers t times to ensure the correctness of data storage. Then, he must pre-compute t verification tokens for each G(j) (j $\in$ {1, . . . , n}), using a PRF f(·), a PRP _(·), a challenge key kchal and a master permutation key KPRP . To generate the i^{th} token for server j, the user acts as follows:

1) Derive a random challenge

2) Compute the set of r randomly-chosen indices:

3) Calculate the token

After token generation, the user has the choice of either keeping the pre-computed tokens locally or storing them in encrypted form on the cloud servers.

K Nearest Neighbor Computation

In the secure k-nearest neighbor (kNN) scheme, Euclidean distance between a database record P_i and a query vector q is used to select k nearest database records.

The secret key is composed of one (d + 1)-bit vector as S and two (d + 1) × (d + 1) invertible matrices as {M1, M2}, where d is the number of fields for each record P_i .

While the computation and communication cost in the query procedure is linear with the number of query keywords in other multiple-keyword search schemes our proposed schemes introduce nearly constant overhead while increasing the number of query keywords. We demonstrate a thorough experimental evaluation of the proposed technique on a real-world dataset: the Enron Email Dataset. We randomly select different number of emails to build dataset. The whole experiment system is implemented by C language on a Linux Server with Intel Xeon Processor 2.93GHz. The public utility routines by Numerical Recipes are employed to compute the inverse of matrix.

The performance of our technique is evaluated regarding the efficiency of two proposed MRSE schemes, as well as the tradeoff between search precision and privacy as a more general search approach, predicate encryption schemes are recently proposed to support both conjunctive and disjunctive search. Conjunctive keyword search returns "all-or-nothing", which means it only returns those documents in which all the keywords specified by the search query appear; disjunctive keyword search returns undifferentiated results, which means it returns every document that contains a subset of the specific keywords, even only one keyword of interest.

ISSN 2348 – 7968

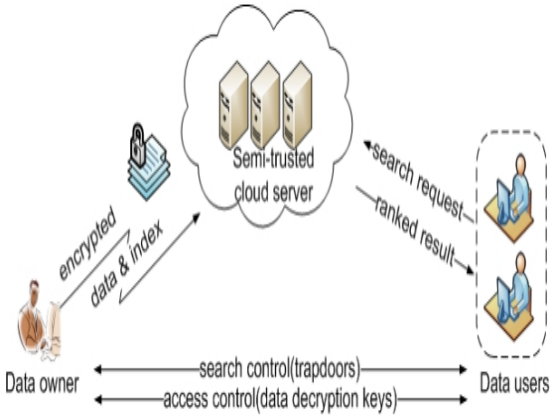

Fig 2 -Architecture of multikeyword search

To enable ranked search for effective utilization of outsourced cloud data under the aforementioned model, our system design should simultaneously achieve security and performance guarantees as follows.

Multi-keyword Ranked Search: To design search schemes which allow multi-keyword query and provide result similarity ranking for effective data retrieval, instead of returning undifferentiated results.

Privacy-Preserving: To prevent the cloud server from learning additional information from the dataset and the index, and to meet privacy requirements specified.

Efficiency: Above goals on functionality and privacy should be achieved with low communication and computation overhead

5. Conclusion

In this paper, for the first time we define and solve the problem of multi-keyword ranked search over encrypted cloud data using Homomorphic token Pre-Computation and establish a variety of privacy requirements. Among various multi-keyword semantics, we choose the efficient similarity measure of "coordinate matching", i.e., as many matches as possible, to effectively capture the relevance of outsourced documents to the query keywords, and use "inner product similarity" to quantitatively evaluate such similarity measure. For meeting the challenge of supporting multi-keyword semantic without privacy breaches, we propose a basic idea of MRSE using secure inner product computation. Then we give two improved MRSE schemes to achieve various stringent privacy requirements in two different threat models. Thorough analysis investigating privacy and efficiency guarantees of proposed schemes is given, and experiments on the real-world dataset show our proposed schemes introduce low overhead on both computation and communication.

6. References

[1] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," ACM SIGCOMM Comput. Commun. Rev., vol. 39, no. 1, pp. 50–55, 2009.

[2] S. Kamara and K. Lauter, "Cryptographic cloud storage," in RLCPS, January 2010, LNCS. Springer, Heidelberg.

[3] A. Singhal, "Modern information retrieval: A brief overview," IEEE Data Engineering Bulletin, vol. 24, no. 4, pp. 35–43, 2001.

[4] I. H. Witten, A. Moffat, and T. C. Bell, "Managing gigabytes: Compressing and indexing documents and images," Morgan Kaufmann Publishing, San Francisco, May 1999.

[5] D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proc. of S&P, 2000.

ISSN 2348 – 7968

[6] E.-J. Goh, "Secure indexes," Cryptology ePrint Archive, 2003, http:// eprint.iacr.org /2003/216.

[7] Y.-C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in Proc. of ACNS, 2005.

[8] R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," in Proc. of ACM CCS, 2006.

[9] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. of EUROCRYPT, 2004.

[10] M. Bellare, A. Boldyreva, and A. ONeill, "Deterministic and efficiently searchable encryption," in Proc. of CRYPTO, 2007.

[11] M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi, "Searchable encryption revisited: Consistency properties, relation to anonymous ibe, and extensions," J. Cryptol., vol. 21, no. 3, pp. 350–391, 2008.

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