www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 4 April 2015, Page No. 11385-11390

Improving Privacy Multi-Keyword Top-K Retrieval Search Over Encrypted Cloud Data

Aashi Qul Huq A[1] Bhaggiaraj S [2]

PG Scholar, Department of(CSE-PG)
Sri Ramakrishna Engineering College,
Anna university, India,
Coimbatore.

Assistant Professor/SG (IT),
Sri Ramakrishna Engineering College,
Anna university, India,
Coimbatore.

ABSTRACT- Cloud computing is the emerging technology, in which storage and retrieval of sensitive data information are increased in usage. The effective data access control should be done in the effective access controlled manner through which one provide an better security. The key work search retrieval over the encrypted cloud data's are complex and tedious process which need to more concerned to improve the user friendly environment while searching for an data over an encrypted data. In this paper, for the first time, we define and solve the challenging problem of privacy-preserving multi-keyword ranked search over encrypted data in cloud computing (MRSE). We establish a set of strict privacy requirements for such a secure cloud data utilization system. Among various multi-keyword semantics, we choose the efficient similarity measure of "coordinate matching", i.e., as many matches as possible, to capture the relevance of data documents to the search query. The proposed methodology proves that the keyword search retrieval is done effectively over the encrypted cloud data's with the consideration of privacy requirements. The experimental results prove that the our proposed methodology is better than the already existing methodologies.

Key words: *Cloud computing, searchable encryption, privacy-preserving, keyword search, ranked search*

1. INTRODUCTION

In cloud computing, data owners may share their outsourced data with a number of users, who might want to only retrieve the data files they are interested in. One of the most popular ways to do so is through keyword-based retrieval. Keyword-based retrieval is a typical data service and widely applied in plaintext scenarios, in which users retrieve relevant files in a file set based on keywords. However, it turns out to be a difficult task in cipher text scenario due to limited operations on encrypted data. Besides, to improve feasibility and save on the expense in the cloud paradigm, it is preferred to get the retrieval result with the most relevant files that match users' interest instead of all the files, which indicates that the files should be ranked in the order of relevance by users' interest and only the files with the highest relevance's are sent back to users. on-demand high-quality applications and services from a shared pool of configurable computing resources [2], [3]. Its great flexibility and economic savings are motivating both individuals and enterprises to outsource their local complex data management system into the cloud. Compared with the preliminary version [1] of this paper, this journal version proposes two new mechanisms to support more search semantics. This version also studies the

support of data/index dynamics in the mechanism design. Moreover, we improve the experimental works by adding the analysis and evaluation of two new schemes. In addition to these improvements, we add more analysis on secure inner product and the privacy part. Preventing the cloud from involving in ranking and entrusting all the work to the user is a natural way to avoid information leakage. However, the limited computational power on the user side and the high computational overhead precludes information security. The issue of secure multi keyword top-k retrieval over encrypted cloud data, thus, is: How to make the cloud do more work during the process of retrieval without information leakage. Cloud computing delivers infrastructure, platform, and software that are made available as subscription-based

services in a pay-as-you-go model to consumers. These services are referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) in industries. The importance of these services was highlighted in a recent report from the University of Berkeley as: "Cloud computing, the long-held dream of computing as a utility has the potential to transform a large part of the IT industry, making software even more attractive as a service".

1.1 Types of cloud services

Cloud Services: ***SaaS (Software As A Service)*** provides all the functions of a sophisticated traditional application to many customers and often thousands of users, but through a Web browser, not a "locally-installed" application. It eliminates customer suspicions about application servers, storage, application development and related, common concerns of IT. Highest-profile examples are Yahoo and Google, and VoIP from Vonage and Skype.

PaaS (Platform as a Service) delivers virtualized servers on which customers can run existing applications or develop new ones without having to worry about maintaining the operating systems, server hardware, load balancing or computing capacity. These vendors provide APIs or development platforms to create and run applications in the cloud – e.g. using the Internet.

IaaS (Infrastructure as a Service) delivers utility computing capability, typically as raw virtual servers, on demand that customers configure and manage. IaaS is designed to replace the functions of an entire data center. This saves cost (time and expense) of capital equipment deployment but does not reduce the cost involved in configuration, integration or management and these tasks must be performed remotely. Apart from these we have the following Cloud computing infrastructure models:

1.2 Need For The Study

In the existing system effective data retrieval needs, the large amount of documents demand the cloud server to perform result relevance ranking, instead of returning un-differentiated results. The large number of data users and documents in cloud, it is crucial for the search service to allow multi-keyword query and provide result similarity ranking to meet the effective data retrieval need. Such ranked search system enables data users to find the most relevant information quickly, rather than burdensomely sorting through every match in the content collection. Ranked search can also elegantly eliminate unnecessary network traffic by sending back only the most relevant data, which is highly desirable in the pay-as you use cloud paradigm. The searchable encryption focuses on single keyword search or Boolean keyword search, and rarely differentiates the search results.

We define and solve the challenging problem of privacy-preserving multi-keyword ranked search over encrypted cloud data (MRSE), and establish a set of strict privacy requirements for such a secure cloud data utilization system to become a reality. Among various multi-keyword semantics, we choose the efficient principle of "coordinate matching", i.e., as many matches as possible, to capture the relevance of data documents to the search query. Specifically, we use "inner product similarity", i.e., the number of query keywords appearing in a document, to quantitatively evaluate such similarity measure of that document to the search query.

2.PROPOSED SYSTEM

We define and solve the challenging problem of privacy-preserving multi-keyword ranked search over encrypted cloud data (MRSE), and establish a set of strict privacy requirements for such a secure cloud data utilization system to become a reality. Among various multi-keyword semantics, we choose the efficient principle of "coordinate matching", i.e., as many matches as possible, to capture the relevance of data documents to the search query. Specifically, we use "inner product similarity", i.e., the number of query keywords appearing in a document, to quantitatively evaluate such similarity measure of that document to the search query.

2. 1Problem Objective

The intent of this research is to identifying the accurate file on the cloud setup file server. The multi keyword ranked search encrypted cloud data (MRSE) are used to collect the files on the system. The information retrieval process are accessed to get the file cloud system. The searchable key encryption are mostly accessed on this research for evaluating to collect the accurate file on the cloud file setup.

2.2 Architecture Diagram

The base station inform the beginning time ,length of time slot ,number of sensor node. Based on these information own energy level of the node is computed and stored. The nodes then select the parent node using EL from its neighbor. Finally collect the information from all sensor node and fuse the information transfer to the base station.

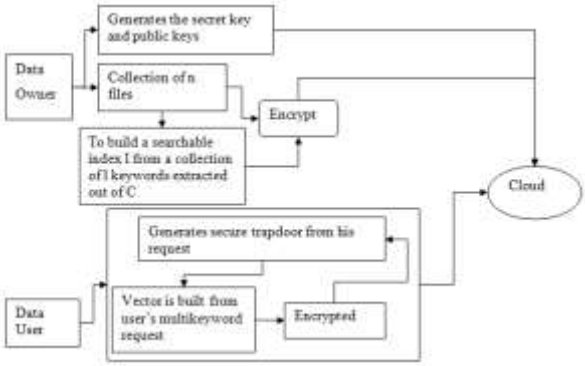

Figure 2

2.3 Advantages

1. Multi-keyword ranked search over encrypted cloud data (MRSE)
2. "Coordinate matching" by inner product similarity.
3. Search result shpuld be ranked by the cloud server according to some ranking criteria.
4. To reduce the communication cost.

3. RELATED WORK

3.1 Single Keyword Searchable Encryption

Traditional single keyword searchable encryption schemes[4], [5], [6], [7], [8], [9], [10], [11], [12], [22], [23] usually build an encrypted searchable index such that its content is hidden to the server unless it is given appropriate trapdoors generated via secret key(s) . It is first studied by Song et al. [4] in the symmetric key setting, and improvements and advanced

security definitions are given in Goh [5], Chang et al. [6], and Carmela et al. [7]. Our early works [22], [23] solve secure ranked keyword search which utilizes keyword frequency to rank results instead of returning undifferentiated results. However, they only supports single keyword search. In the public key setting, Boneh et al. [8] present the first searchable encryption construction, where anyone with public key can write to the data stored on server but only authorized users with private key can search. Public key solutions are usually very computationally expensive however. Further more ,the keyword privacy could not be protected in the public key setting since server could encrypt any keyword with public key and then use the received trapdoor to evaluate this cipher text.

3.2 Boolean Keyword Searchable Encryption

To enrich search functionalities, conjunctive keyword search [13], [14], [11], [16], [17] over encrypted data have been proposed. These schemes incur large overhead caused by their fundamental primitives, such as computation cost by bilinear map, for example, [15], or communication cost by secret sharing, for example, [15]. As a more general search approach, predicate encryption schemes [18], [19],[20] 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. In short, none of existing Boolean keyword searchable encryption schemes support multiple keywords ranked search over encrypted cloud data while preserving privacy as we propose to explore in this paper.

.

4 IMPLEMENTATION PHASES

1. Create Cloud Setup
2. Initialization
3. Retrieval

4.1CREATE CLOUD SETUP

A simulation toolkit enables modeling and simulation of Cloud computing systems and application provisioning environments. The Cloud Sim toolkit supports both system and behavior modeling of Cloud system components such as data centers, virtual machines (VMs) and resource provisioning policies. It implements generic application provisioning techniques that can be extended with ease and limited effort. Currently, it supports modeling and simulation of Cloud computing environments consisting of both single and inter-networked clouds (federation of clouds). Moreover, it exposes custom interfaces for implementing policies and provisioning techniques for allocation of VMs under inter-networked Cloud computing scenarios. In this module we are creating cloud users and datacenters and cloud virtual machines as per our requirement. The term instance type will be used to differentiate between VMs with different hardware characteristics.

4.2 INITIALIZATION:

In the initialization phase the security parameters for secured search over network is initialized. The initialization phase consists of two stages. The Setup stage involves the secure initialization, while the Index Build stage involves operations on plaintext. For security concerns, the vast majority of work should only be done by the data owner.

Setup Phase:

A. The data owner calls KeyGen(λ) to generate the secret key SK and public key set PK for the homographic encryption scheme. Then the data owner assigns SK to the authorized data users.

B. The data owner extracts the collection of keywords, W ={ w_1; w_2; . . . ; w_l}, and their TF and IDF values out of the collection of n files, C ={f_1; f_2; . . . ; f_n}. For each file f ∈ C, the data owner builds a (l +1)-dimensional vector v_i ={id_i; $t_{i;1}$; $t_{i;2}$; . . . ; $t_{i;l}$}, where $t_{i;j}$ ={ tf-idf $_{wj;fi}$ ($1 \leq j \leq l$)} The searchable index I ={v_{ij} $1 \leq j \leq n$ }.

Build Index Phase:

C. The data owner encrypts the searchable index I to secure searchable index I` = $\{v_i' | 1 \leq j \leq n\}$

D. The data owner encrypts C={f_1; f_2; . . . ; f_n} into $C' = \{f_1', f_2', \ldots, f_n'\}$ with other cryptology schemes, and then outsources C' and I' to the cloud server.

4.3 RETRIEVAL

The Retrieval phase involves Trapdoor Gen, Score Calculate, and Rank, in which the data user and the cloud server are involved. As a result of the limited computing power on the user side, the computing work should be left to server side as much as possible. Meanwhile, the confidentiality privacy of sensitive information cannot be violated. The ranking should be left to the user side while the cloud server still does most of the work without learning any sensitive information.

- TrapdoorGen(REQ; PK). The data user generates secure trapdoor from his request REQ. Vector T_ω is built from user's multikeyword request REQ and then encrypted into secure trapdoor $T_{\overline{\omega}}$ with public key from PK, output the secure trapdoor $T_{\overline{\omega}}$
- Score Calculate($T_{\overline{\omega}}$, I'). When receives secure trapdoor $T_{\overline{\omega}}$ the cloud server computes the scores of each files in I' with $T_{\overline{\omega}}$ and returns the encrypted result vector $\aleph$ back to the data user.
- Ranking($\aleph$, SK, k). The data user decrypts the vector $\aleph$ with secret key SK and then requests and gets the files with top-k scores.

Limited computing power on the user side, we are mostly concerned about the complexity of ranking. Since the decryption of $\aleph$ can be accomplished in O(n) time, the only function that could influence the time complexity of ranking is the top-k select algorithm, i.e., TOPKSELECT algorithm. The details of TOPKSELECT algorithm are shown in algorithm 1. Since the complexity of the INSERT algorithm is O(k) the overall complexity of TOPKSELECT algorithm is O(nk). Note that k, which denotes the number of files that are most relevant to the user's interest, is generally very small compared to the total number of files. In case of large value of k, the complexity of the TOPKSELECT algorithm can be

easily reduced to O (n log k) by introducing a fixed-size min-heap.

Algorithm 1 TOPKSELECT(source, k)

Input:
list source to be selected
number k
Initialization:
Set topk ←Ø; topkid ←Ø;
Iteration:
1: for all item ϵ source do
2 INSERT (topk, (item, itemindex))
3: end for
4: for all tuple ϵ topk do
5 topkid.append(tuple[1])
6: end for
Output:
topkid

Algorithm 2 INSERT(topk, (item, itemindex))

Input:
list topk to store the top-k scoring item
tuple (item, itemindex)
Iteration:
1: if len(topk) < k then
2: insert (item, itemindex) into topk in nondecreasing order of item
3: else
4: for all element ϵ topic do
5: if item < element[0] then
6: continue
7: else
8: discard topk[0], insert (item, itemindex) into topk in nondecreasing order of item
9: end if
10: end for
11: end if

5. EXPERIMENTAL ANALYSIS

5.1 QUERY TIME

The time taken to submit an query with difference workload sizes are evaluated and compared with the existing algorithms. The below graph (figure 5.1)shows the comparison of MRSE 1 and MRSE 2 algorithm with different file sizes.

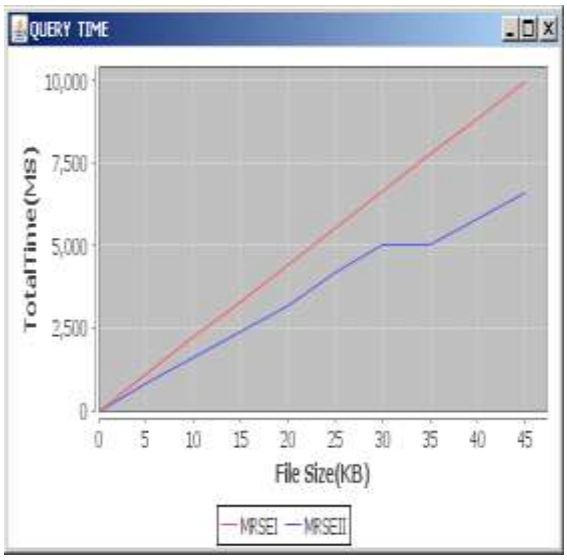

Figure 5.1 Query Time Comparison

5.2 COMPUTATION OVERHEAD

The computation overhead defines the overall processing capacity utilized for process the user submitted query. The computation overhead should be minimized in order to improve the overall effectiveness of the proposed method(figure 5.2)

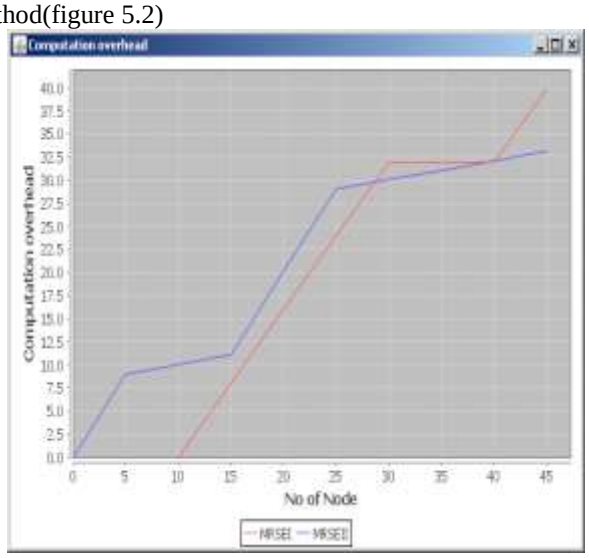

Figure 5.2 Computation Overhead Comparison

5.3 STORAGE OVERHEAD

The storage overhead defines, the amount of space consumed for storing the data owner's encrypted data into the cloud servers. It is ratio between the time complexity taken to store the data and retrieve the stored data (figure 5.3).

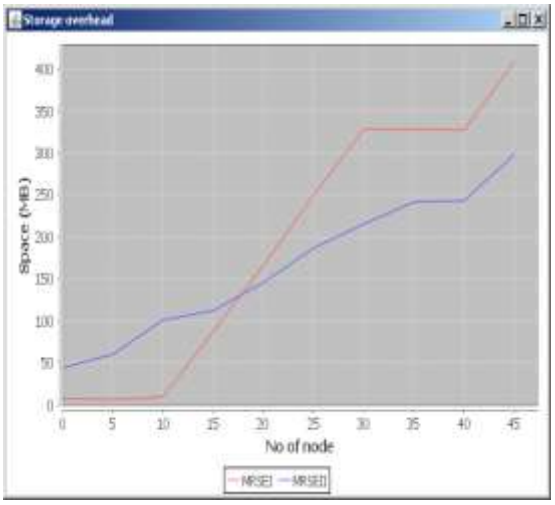

Figure 5.3 Storage Overhead

6. Conclusion and Future Enhancement

We motivate and solve the problem of secure multi keyword top-k retrieval over encrypted cloud data. In this work, a new framework is proposed for the problem of multi-keyword ranked search over encrypted cloud data, and to establish a variety of privacy requirements. Among various multi-keyword semantics, the efficient similarity measure is "coordinate matching", i.e., as many matches are 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, MRSE framework is proposed using secure inner product computation. Thorough analysis investigating privacy and efficiency guarantees of proposed schemes is given, and experiments on the real-world dataset shows our proposed scheme introduces low overhead on both computation and communication.

6.1Future Work

Future effort will be focused on further reducing the time needed for indexing calculations by optimizing the encoding algorithm. Moreover, the presented methodology will be expanded to 3-D to support both location and intensity related query. In our future work, we will explore checking the integrity of the rank order in the search result assuming the cloud server is untrusted. The existing method selects the parent node based on only the residual energy level. So the drawback in this method is if the parent node has less communication capacity, high interference and congestion there will be less network performance in terms of packet delivery ratio, delay, throughput etc. So, this can be considered in future work.

7.REFERENCES

[1]. R. Curtmola, J.A. Garay, S. Kamara, and R. Ostrovsky, "Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions," Proc. ACM 13th Conf. Computer and Comm. Security (CCS), 2006.

[2] 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.

[3] N. Cao, S. Yu, Z. Yang, W. Lou, and Y. Hou, "LT Codes-Based Secure and Reliable Cloud Storage Service," Proc. IEEE INFOCOM,
pp. 693-701, 2012.

[4] D. Song, D. Wagner, and A. Perrig, "Practical Techniques for Searches on Encrypted Data," Proc. IEEE Symp. Security and Privacy, 2000.

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

[6] Y.-C. Chang and M. Mitzenmacher, "Privacy Preserving Keyword Searches on Remote Encrypted Data," Proc. Third Int'l Conf. Applied Cryptography and Network Security, 2005.

[7] R. Curtmola, J.A. Garay, S. Kamara, and R. Ostrovsky, "Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions,"
Proc. 13th ACM Conf. Computer and Comm. Security(CCS '06), 2006.

[8] D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public Key Encryption with Keyword Search," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2004.

[9] M. Bellare, A. Boldyreva, and A. ONeill, "Deterministic and Efficiently Searchable Encryption," Proc. 27th Ann. Int'l Cryptology
Conf. Advances in Cryptology (CRYPTO '07), 2007.

[10] 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. Cryptology, vol. 21, no. 3, pp. 350- 391, 2008.

[11] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy Keyword Search Over Encrypted Data in Cloud Computing," Proc. IEEE INFOCOM, Mar. 2010.

[12] D. Boneh, E. Kushilevitz, R. Ostrovsky, and W.E.S. III, "Public Key Encryption That Allows PIR Queries," Proc. 27th Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO '07), 2007.

[13] P. Golle, J. Staddon, and B. Waters, "Secure Conjunctive Keyword Search over Encrypted Data," Proc. Applied Cryptography and Network Security, pp. 31-45, 2004.

[14] L. Ballard, S. Kamara, and F. Monrose, "Achieving Efficient Conjunctive Keyword Searches over Encrypted Data," Proc. Seventh Int'l Conf. Information and Comm. Security (ICICS '05),
2005.

[15] D. Boneh and B. Waters, "Conjunctive, Subset, and Range Queries on Encrypted Data," Proc. Fourth Conf. Theory Cryptography (TCC),
pp. 535-554, 2007.

[16] R. Brinkman, "Searching in Encrypted Data," PhD thesis, Univ. ofTwente, 2007.

[17] Y. Hwang and P. Lee, "Public Key Encryption with Conjunctive Keyword Search and Its Extension to a Multi-User System,"Pairing, vol. 4575, pp. 2-22, 2007.

[18] J. Katz, A. Sahai, and B. Waters, "Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products," Proc. 27th Ann. Int'l Conf. Theory and Application of Cryptographic
Techniques (EUROCRYPT), 2008.

[19] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully Secure Functional Encryption: Attribute-Based Encryption and (Hierarchical) Inner Product Encryption," Proc. 29th Ann.Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '10), 2010.

[20] E. Shen, E. Shi, and B. Waters, "Predicate Privacy in Encryption Systems," Proc. Sixth Theory of Cryptography Conf. Theory of Cryptography (TCC), 2009.

[21] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized Private Keyword Search over Encrypted Data in Cloud Computing," Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '10), pp. 383-392, June 2011.
[22] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure Ranked Keyword Search over Encrypted Cloud Data," Proc. IEEE 30th Int'l Conf. Distributed Computing Systems (ICDCS '10), 2010.
[23] C. Wang, N. Cao, K. Ren, and W. Lou, "Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 8, pp. 1467-1479, Aug. 2012.