IJCSE International Journal of Computer Sciences & Engineering Open Access
Review Paper Volume-3, Issue-7 E-ISSN: 2347-2693

Loss Less and Privacy Preserved Data Retrieval in Cloud Environment Using TRSE

T. Kavitha* and P. Nageswara Rao

[1*,2]Dept. of CSE, Swetha Institute of Technology And Sciences, Tirupati, AP, INDIA

Received: Jun/22/2015 Revised: Jul/06/2015 Accepted: July/24/2015 Published: July/30/ 2015

Abstract: In the modern era of the computing world, the data producing and using it is becoming large and instant at various places. For availing the data at different locations for processing we need to store it in the global platform. Cloud environment provides a best and easy way for this. Cloud computing is becoming as the essential thing for high-quality data services. However there are some potential problems with respect to data security. Here encryption techniques can be used for providing security, but with restricted efficiency. In this paper we propose a new encryption mechanism for providing data security in cloud environment. We propose a two round searchable encryption which supports multi keyword retrieval. Here we adapted a vector space model for improving search accuracy; the elgamal encryption technique allows users to involve in the ranking, while the essential key part of encryption will be done at the source itself. The proposed improves the data security and reduces data leakage.

Key words: Cloud server, Data security, Structure strength, Resemblance matching, Vector model

1. INTRODUCTION

In the global technical world, the amount of data producing and storing is increasing from day to day. Especially small companies can't establish the hardware and infrastructure for storing and maintaining the data. Another drawback with the traditional data storage mechanism is that: you need to have the complete infrastructure and hardware where you stored your data, to have access to that data.

The cloud environment provides solution to both of these problems, cloud service providers allow you, to store your information in the cloud and you can get access to that data from any place at any time, with the help of internet connection. The cloud services are become mandatory for the outsourcing of data. But the problem of loss of information and security issues [1], [2] are still with the cloud computing [3] mechanism.

The root cause for less security of data is the cloud itself. When users store their data such as e-mails, company details, personal information the cloud service providers are establishing the connection between the users and the data from different locations and at any time. This leads to uncertainty in the data loss and burden for the service providers to keep track of the user's actions on the cloud data [4]. For providing security for the data in general the cloud vendors encrypt the original data before storing it in the cloud environment, by which effective utilization of data can be done. Even the data

Encryption can give security, vendors still need to connect to the cloud which leads to loss if information. Owners of the cloud stored data may wish to provide access to the stored information to several users. The best solution for this is the keyword-based searching mechanism. To get the relevant information based on the users search better to give ranking to the files in the cloud environment, by which users can get the related files based on their search.

A variety of symmetric encryption mechanisms were proposed for data search in the cloud environment. Traditional mechanism only support only Boolean keyword search [5], [6]. To provide more security for the cloud data without being altered the efficiency, top-k single keyword retrieval methods were used [7], [8], [9]. Later top-k multi keyword search methods were also used [10], [11], but it was suffered from the problem of poor security and efficiency.

Burden on the cloud can be reduced by avoiding the ranking user's files by cloud, which can be given to the users. With this information loss can be reduced. Hence the secure multi keyword retrieval mechanism is all about how to get the users interested information with more relevance, by reducing the burden on the cloud, without being loss of data.

In the paper, we introduce the concept of resemblance matching and structure strength to solve the issue of security. The security problem is solved by using the two round searchable encryption mechanisms. The concepts of resemblance matching and structure strength were used to provide security by which, privacy issues were violated. The two round searchable encryption were helped in providing multi keyword retrieval.

2. FUNDAMENTAL CONCEPTS:

Consider a basic model of cloud computing environment. There will be an owner who used to store

his information in the cloud, the cloud server, which provides storage and services to the owner and authenticated users and finally the authenticated third party. The cloud server provides services to the authenticated third party (ATP), it gives access to the owners information depending on the type of permissions he is given. Sometimes any kind of data loss will be treated as security issue which is not acceptable.

The owner may be stored a number of *m* data files $\{f_1, f_2, f_3, \ldots, f_n\}$ of his own interest in the cloud environment. Owner will use an encryption mechanism to outsource the files onto the cloud environment. The cloud environment should provide a keyword based search for the authenticated third party (ATP). The owner has to maintain an index for the list of files he stored in the cloud. For this gather some keywords *p* from the files K= $\{x_1, x_2, \ldots, x_p\}$, now give the access rights to the ATP on both encrypted files and the index generated from K.

The ATP can get access to the data files just by providing a query to the cloud which will be evaluated at the cloud environment based on the keywords he has given. The appropriate files will be sent to the ATP in encrypted form, which he can decrypt and use the information.

Similarity Counting

Most of the multi keyword encryption mechanisms support only Boolean queries, i.e., whether the keywords you are searching are there in the file or not. But it gives poor performance, keeping in mind this issue it should give the results by considering the similar keywords in the file, based on the ATP queried keywords.

Counting is a measure for similarity. Data files in the cloud environment can be ranked based on the similarity count. Several methods are there for calculating the similarity counting.

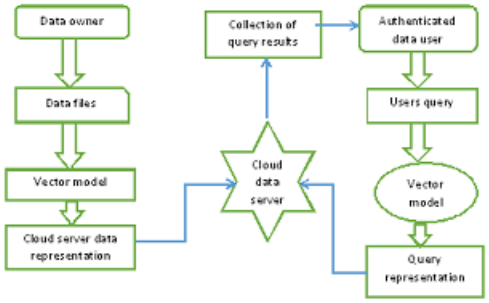

Fig: System Architecture

The widely used one is that, frequency calculation mechanism, which involves two attributes word frequency and file frequency. Word frequency ($wf_{w,f}$) is the number of occurrences of a word *w,* in file *f.* file frequency (Ff_w) is the number of files that contain particular word. The $wf\text{-}idf_{w,f}$ weighting scheme assigns to a word *w* in a file *f* is given by $wf\text{-}idf_{w,f} = wf_{w,f} * idf_w$. with this similarity count mechanism the weights of frequently occurring words are reduced whereas, weights of the words which occr rarely are increased.

The Vector Model

The *wf-idf* gives the weights of single keywords in a file, now we need to apply this for multi keyword. The vector model [12], is an algebraic model which has number of dimensions. Each dimension is related to a separate word if interest. If the particular keyword is exist in the file its value in the vector model is a nonzero, if it is not found the dimension will have zero. The vector model allows computing a continuous similarity between files and the queries, which really meets our top-k retrieval mechanism. A query is also represented as a vector. Given the counts of the keywords, files can be assigned ranks and most similar files can be retrieved.

3. PROBLEM DEFINITION

Consider the cloud server which is truthful and probing [7], the cloud server will certainly follow the mechanism adapted but it is probing to analyze the information and the queries to gather more information.

Statistical Data Loss

Even though all the data files and ATP queries are encrypted before they are processed by the cloud server, the cloud environment still process the data files to gather some additional information. We denote the data loss as statistical data loss (SDL). There may be two possible drawbacks: *word distribution* and *interdistribution.* The word distribution of a word *w* is the frequency distribution of *w* on the file. Interdistribution is the frequency of count of each word l $(l \in f)$. These two can be computed either form the cipher text or through statistical analysis. Here the interesting patterns retrieved to which the files and the queries are searched are the keyword in both the requests is same or not.

It is observed that the similarity between words and files is given by the distribution information. Obviously words with similar kind of distribution will give multi keyword search mechanism a better result. For instance the labels in both *yahoo-mail* and *Gmail* will be similar. It should be noted that files with same interdistribution will always come under the same category.

***K*-Resemblance Matching**

To restrict the information loss at the cloud server environment methods [7], [8] were deployed to distribute similarity counts. These methods however concentrate on distribution of individual word or file,

discarding the relevance between them which leads to security problem. Hence the hence the k-feature matching is employed to address this problem.

Here we compute resemblance matching among two words, the words are said to be co-occur if there resemblance matching, suppose if $k_{i,j}$ =0.5 means the word *I* occur in half of the files that *j* appears. We keep a threshold for the resemblance. The two words are said to be matched if $k>=k_0$. Sometimes there may be a one to many kind of mapping occur in the word similarity. The order preserving strategy will help in giving the relevant keywords based on the top-k keyword matching mechanism.

Structure Strength

Information security in the cloud environment can be handled by using the resemblance matching. The co-occurrence [13] of words is the basic thing we need to consider in this mechanism. The co-occurrence between two words can be computed by means of various mechanisms but not only these information retrieval and *w*-count.

If the encryption mechanism is not implemented properly the privacy policy of the data stored in the cloud may be affected. Consider two words w_1 and w_2, we can say these two co-occur with each other most of the times in a particular file then $k_{w1,w2}=1$. Structure strength is specifically for words; hence it should be hidden from the cloud environment. The order-preserving top-k keyword search will not conceal about the structure strength, and it requires server side computations on the cipher text, to be order-preserving. Hence it is insecure for statistical data loss. Hence it is not good that giving the ranking mechanism to the cloud environment completely.

4. IMPLEMENTATION

• **Setup** (Z): Both the public key and secret keys are generated by the data owner for the homomorphism encryption scheme. Z is a parameter that is taken as input; secret key and a public key are taken as output parameters.

• **Index Build** (C,PK): Secure searchable index from the file is given by the data owner. Technologies from International research community like stemming are employed to build searchable index I from collection of files, and then I is encrypted into I(encrypted form) with private key, distribute the secure searchable index I.

• **TrapdoorGen** (REQ, PK): the secure trapdoor is generated by the ATP from his request. Vector is built by the request of multi-keyword and then encrypted with public key from PK to secure trapdoor T, output the secure trapdoor T.

• **Count Calculate** (T, I): When it receives secure trapdoor T generated by the ATP, the cloud server computes the priority of each item in I with T and displays the encrypted form of indexed result vector to the data users display.

• **Rank** (SK,k): The vector is decrypted by the data user with secret key SK, and then user requests and returns the data with top-k priority.

Key Generation

• KeyGen (λ): The secret key SK is an odd η-bit number randomly selected from the interval [2η−1, 2η]. The set of public keys PK = {k0, k1, ..., kτ} {pq+r|q [0, 2γ/p], r, 2Z ∩(−2ρ, 2ρ)} and ρ denotes the bit length of r. The noise factor x is randomly selected from the interval (22μ, 22(μ+1)], where μ denotes the bit length of atomic plaintext. Note that the secret key is used for encryption and the public keys are used for decryption, which are different from the concepts of keys in public-key cryptography.

Encryption:

• Encrypt (PK,m): Randomly choose a subset R {1, 2, ..., τ} and an integer r (−22ρ, 22ρ), and then return cipher text c = m + xr_ +iR ki.3

Evaluation:

• Evaluate (c1, c2, ..., ct): Implement the binary multiplication and gates to the cipher text ci, perform all the intended operations, and then display the resulting data χ.

Decryption:

• Decrypt (p, χ): Output m_ = (χ mod p) mod x

Here ρ = λ, η = O(λ2), γ = O(λ5). The relatively time-consuming scheme is FHEI. so we use it only for the encryption of the searchable index I, while the file set C can be encrypted with other symmetric encryption scheme. Note that the Evaluate stage sets no limit to how many addition or multiplication operations can be executed without encryption. In fact, the cipher text of an integer, which is another integer, can be applied as many evaluations as needed.

Algorithm working

The key generation algorithm works as follows: Alice generates a group of multiplicative cycle with the required properties.

• Alice will consider a random value *p* from {0, . . . , *q-1*).

• Calculate H= d^p.

• Now it publishes H as the public key and p as the private key which must be kept secret.

This encryption algorithm can be applied to any cloud data which is cyclic in nature. Its privacy depends on the complexity of problem related mathematical calculations. The encryption algorithm consists of three mechanisms: the key generator, the encryption algorithm, and the decryption algorithm.

The encryption algorithm workings as follows: on the way to encrypt a message to Alice under its public key,

• Bob chooses a random *r* from {0, . . . , *q-1*) , then calculates $A_1= g^r$.

• Calculates the shared secret H^r.

• Now bob converts its secret message M into a new form M^l of G.

• Calculates $A_2=M^l*S$.

• Bob sends the cipher text $(A_1,A_2)=(g^r,M^l*H^r)$ to Alice.

Here the ATP can easily find h^r if he knows the value of M^l. Hence a new *r* is generated for each data message for providing more security. The decryption algorithm works as follows: to decrypt a cipher text with her private key *p*,

• Alice calculates the shared secret $s=A_1^p$

• Now computes $M^l=A_2*s^{-1}$which she then converts back into the plaintext message M, where s^{-1} is the inverse of sin in the group G.

5. RELATED WORK

As the cloud storage is the widely used and emerging area for the outsourcing of information privacy mechanism should be employed with greater interest. Variety of encryption mechanisms [5], [6], [14] was deployed in the traditional searchable symmetric encryption, which majorly concentrate on Boolean data values. A new ranking mechanism [9] was also proposed for multi keyword based search in the cloud environment. Another author proposed top-k retrieval over encrypted data. Some existing mechanisms propose various schemes which support Boolean multi keyword search [7]. Most of the authors they fail to address the loss of data and privacy preservation in the server side. We, therefore, concentrated on addressing these issues.

6. CONCLUSION

In this paper, we employed a new encryption technique for providing cloud data privacy. We define resemblance matching and structure strength which will concentrate on data loss. We then proposed a two round searchable encryption mechanism which addresses the security problem of the cloud data. The multi keyword top-k retrieval is possible with more efficiency with this new mechanism. Hence the security is improved for the owner's data in the cloud.

REFERENCES

[1] M. Arrington, "Gmail Disaster: Reports of Mass Email Deletions,"*http://www.techrunch.com/2006/12/28/gmail-disasterreports-of-mass-email-deletions/, Dec.**2006***.

[2] Amazon.com, "Amazon s3 Availability Event: July 20, **2008**," *http://status.aws.amazon.com/s3-20080720.html, 2008.*

[3] M. Arumbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, "A View of Cloud Computing" *Comm. ACM, vol 53, no.4,pp.50-58, 2010.*

[4] C. Leslie, " NSA Has Massive Database of Americans, Phonecalls,"*http://usatoday30.usatoday.com/news/washington/ 2006-05-10/, **2013***.

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

[6] D. Boneh, G. Crescenzo, R. Ostrovsky, and G. Persiano, " Publickey Encryption With Keyword Search*," proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (Eurocrypt),* **2004**.

[7] 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),* **2010**.

[8] S. Zerr, D. Olmedilla, W. Nejdl, and W. Siberski, "Zerber+r: top-k Retrieval from a Confidential Index," *Proc. 12th INT'l Conf. Extending Database Technology: Advances in Database Technology (EDBT), **2009***.

[9] A .Swaminathan, Y. Mao, G.M. Su, H. Gou, A.L. Varna, S. He, M. Wu, and D.W. Oard, "Confidentiality Preserving Rank-Ordered Search," *Proc. Workshop Storage Security and Survivability, **2007***.

[10] N. Cao, C. Wang, M. Li, K. Ren, and W.Lou, "Privacy-Preserving MultiKeyword Ranked Search over Encrypted Cloud Data," *Proc. IEEE INFOCOM*, **2011**.

[11] H. Hu, J. Xu, C. Ren, and B. Choi, " Processing Private Queries over Untrusted Data Cloud through Privacy Homomorphism," *Proc. IEEE 27th INt'l Conf. Data Eng. (ICDE), **2011***.

[12] D. Dubin, "The Most Influential Paper Gerard Salton Never Wrote," *Library Trends,* vol. 52, no. 4, pp. 748-764, **2004**.

[13] S. Gries, " Useful Statistics for Corpus Linguistics," *A Mosaic of Corpus Linguistics: Selected Approaches,* Aquilino Sanchez Moises Almela, eds., pp. 269-291, Peter Lang, **2010**.

[14] R. Curtmol, 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***.

About Authors

Ms. T. Kavitha completed her Bachelor's Degree from JNTU, Hyderabad. Now she is pursuing her Master's degree from JNTUA. Her areas of interest include computer networks, mobile computing, cloud computing.

Mr. P. Nageswara Rao compelted Master's degree from Acharya Nagarjuna University. Currently he is working as HOD, CSE and PRINCIPAL at SITS, Tirupati. His areas of interest include computer networks, artificial intelligence, cloud computing. He has published several papers in various journals.