Advanced Science and Technology Letters
Vol.139 (CCSC 2016), pp.443-449
http://dx.doi.org/10.14257/astl.2016.139.88

Research on the Method of Cloud Computing Storage Security based on the Homomorphic Encryption Method

Jun Wu, Jing Chen

Yiwu Industrial & Commercial College, Yiwu,
322000 Zhejiang province, China
{Jun Wu} 78549608@qq.com

Abstract. Based on the idea of the encryption algorithm, a new method for data storage in cloud computing is designed. Experimental results show that the proposed method in data encryption, data decryption, data retrieval and other aspects of the implementation of the efficiency is significantly higher than the BGV algorithm, and more suitable for the safe storage of cloud data.

Keywords: cloud storage, Homomorphic Encryption, data retrieval, inverted sort

1 Introduction

Cloud computing provides on-demand services for cloud computing users by using virtualization technology and distributed computing technology to build a huge pool of resources to integrate computing resources in the network idle [1]. Cloud users only need to purchase or lease the computing resources they need without having to care about the sources and management of these resources. It freed from the pressure of the cloud users from infrastructure management and maintenance work, so that users can focus more on their core business [2]. Therefore, more and more enterprises, institutions choose cloud computing services as its information resource management service providers and cloud computing services has become an important business in the field of IT. IBM, Amazon, Google, Microsoft and other companies have also launched their own cloud computing service [3-4].

In order to promote the development of cloud computing faster and better, cloud computing security issues have to be effectively resolved. At present, the research institutions, government organizations and cloud service providers have invested a lot of manpower and material resources to study and solve the security problem of cloud computing [5]. In order to promote the healthy and rapid development of cloud computing and cloud storage services, and to provide safe and effective personal information service for individuals, businesses and society, various research institutions, enterprises have put forward cloud security solutions or suggestions [6]. Kumar proposed a comprehensive cloud computing security framework. The framework includes three parts, cloud security objectives, cloud security services and cloud security evaluation system. Among them, the cloud security service system is

ISSN: 2287-1233 ASTL

the most critical, from three levels of infrastructure services, basic services and application services to ensure the security of cloud computing [7]. Ryoo proposed secure cloud architecture. The framework uses isolation mechanism, encryption mechanism and VPN channel mechanism to isolate store the data in the cloud, and establish a virtual private channel for user communication to ensure the security of data transmission [8].

The cloud security storage model based on encryption storage ensures safe storage of data. However, due to the data is stored in the cloud data center, which set up obstacles for the use of cloud data. In order to facilitate the user to retrieve ciphertext in the cloud, the researchers propose a search encryption mechanism, which supports the data encryption algorithm [9]. The data in the cloud is stored in the encrypted data, and the decryption key is saved by the data of the main users, avoiding the risk of data leakage, but it brings a lot of inconvenience to the cloud. To this end, the researchers proposed the concept of proxy re encryption to solve the problem of sharing [10].

In this paper, based on the research of the cloud storage security, we use the theory and idea of the encryption to design a new algorithm and model in cloud data encryption and data retrieval.

2 Design of Data Encryption Algorithm

The first step, set the parameter selection set, as shown in the formula (1):

$$P(1^{\lambda},1^{L}):P=\{P_1,\cdots,P_j,\cdots,P_L\} \tag{1}$$

Here, P_j is used to represent the parameters of each layer, and can be expressed as follows:

$$P_j=\{r_L,\cdots,r_0,\chi_j,e_j,M\} \tag{2}$$

Here, from r_L to r_0 represent a decreasing mode sequence, χ_j represents the distribution which the ring dimension is e_j , and the calculation of M is $M=\lceil(2n+1)\log r\rceil$.

Second step, generating key $K(P)$ .The key generation algorithm is divided into two parts. First, generate the encryption and decryption key T_j and P_j for each layer. Secondly, combinate the encryption and decryption keys that generated by each layer to generate algorithm encryption and decryption key T_k and P_k .

The encryption and decryption key of each layer generate $T_k\in R_q^2$ and $P_k\in R_q^{N\times 2}$, here the expression of R is as follows:

$$R = \frac{Z[x]}{x^e + 1} \tag{3}$$

Here, e is the power of 2.

Encryption and decryption key generation algorithm is as follows:

$$\begin{cases} T_k = (T_0, \cdots, T_L) \\ P_k = (P_0, \cdots, P_L, \tau(T_1' \to T_0), \cdots, \tau(T_L' \to T_{L-1})) \\ T_j' = T_j \otimes T_j \\ \tau(T_{j+1}' \to T_j) = swithK(T_{j+1}', T_j) \end{cases} \tag{4}$$

The third step, the processing of the encryption algorithm is as follows:

$$\begin{cases} E(P, P_k, n) : d = n + P_L^T \\ n \leftarrow (n, 0) \\ r \leftarrow R_2^N \end{cases} \tag{5}$$

The fourth step, the processing of the decryption algorithm is as follows:

$$E(P, P_k, d) : n^* = ((< d, T_j > \bmod_T) \bmod 2) \tag{6}$$

The fifth step, the design of the computation algorithm of the ciphertext is as follows.

$E(P_k, g, d_1, d_2, \cdots, d_n)$ is used to express the ciphertext, the result D_g obtained from the computation of the ciphertext $d_1, d_2, \cdots, d_n$ in the operation of the function g is equal to the plaintext result N_g generated by the plaintext $n_1, n_2, \cdots, n_n$ in the operation of the function g after decryption. That is :

$$\begin{aligned} D_g &= g_{P_k}(d_1, d_2, \cdots, d_n) \\ &= g(n_1, n_2, \cdots, n_n) \\ &= N_g \end{aligned} \tag{7}$$

Encryption algorithm is the core of the whole fully homomorphic encryption algorithm, so the ciphertext computation algorithm is the most complex. The most complex algorithm in the ciphertext computation algorithm is the operation function $g = \{add, multi, refresh\}$. In the function, add represents the add operation, $multi$ represents the multiply operation, and the $refresh$ represents the update operation.

In the fully homomorphic encryption scheme, both the ciphertext and the key are vectors, the definition of the ciphertext product is a tensor $d_i \otimes d_j$, the corresponding key is $T \otimes T$, so the product will lead to the rapid growth of the ciphertext dimension, $refresh$ is a method of reducing the noise by using the key exchange and the mode switching technology to reduce the dimension of the cipher text to the original dimension.

3 Design of Data Retrieval Algorithm

3.1 Inverted index

Index file is the core technology of full-text retrieval. At present, the most widely used full-text indexing model is inverted index. Inverted index, also call reverse index, is a kind of index structure which is use key words as the index keys and the list access portal, and used to store the mapping of a keyword in a document or a set of documents in a full text search.

The inverted index structure is composed of index files and inverted files, and the index file is made of entries and record data which is consisted of logical record pointer. Logical record pointer point to logical address of the inverted file. The inverted file contains information about the document that contains the corresponding entry, mainly about document address, word frequency and entry position.

Inverted index structure is based on plaintext retrieval, the indexed words are stored in plain text, to facilitate the attacker to analyze the text.

For security reasons, in this paper, we design the inverted index structure of the ciphertext as shown in Figure 1.

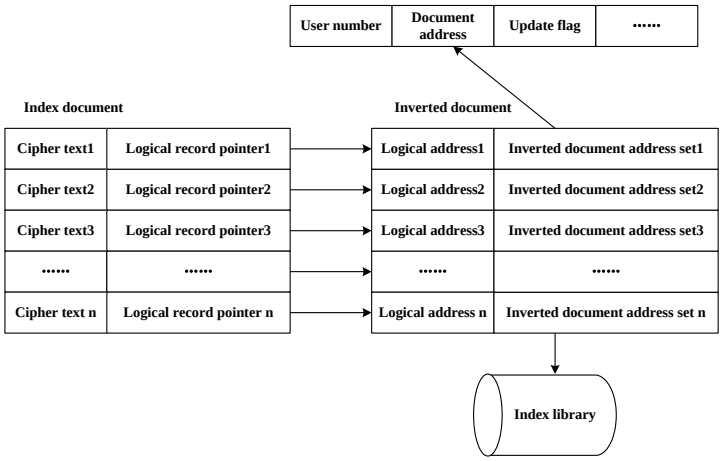

Fig.1. The inverted sort structure

3.2 Ciphertext Scanning

The ciphertext scanning is to query the specified key from the ciphertext set. If the key is in the document, then record the related information of the key words like address, frequency. Ciphertext linear search algorithm is usually used to assume ciphertext scanning, and it can be judged whether the key word is in the query document and the number of times by means of a pair of one to one cipher key word matching.

The linear search algorithm $G_d(T,E)$ of the ciphertext is a linear comparison of the cipher text, to determine whether include the query keywords T. However, the general encryption algorithm does not support linear computation, so we should construct the linear mapping function $U_T = Trap(T,E)$ of the query keywords before the ciphertext retrieval, and then execute linear query.

Encryption algorithm is different, and its key word linear mapping function is also different, so the process of linear retrieval of the text is more complex. The cipher text retrieval process based on the encryption mechanism is more convenient because homomorphic encryption support ciphertext calculation, and plaintext linear search algorithm $G(T,nE)$ can generate the ciphertext linear search algorithm $G_d(T_d,E)$ by the calculation of ciphertext algorithm $Eval()$, so as to complete the calculation of the key words T.

4 Experimental Results and Analysis

In order to verify the effectiveness of the cloud storage security method based on the encryption algorithm proposed in this paper, next, expand the experimental study., we choose the most ideal BGV algorithm as the contrast algorithm.

First of all, test the effect of the two algorithms in data encryption. The encryption key space is gradually increased from 100MB to 700MB according to the step size is the speed of 100M, and the time of data encryption is compared between the two algorithms. The results of the comparison curve are shown in Figure 2.

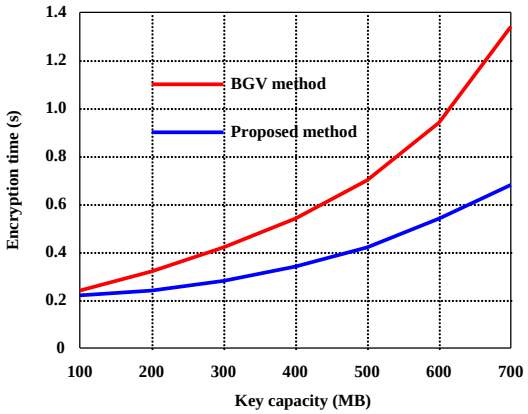

Fig.2. comparison result of data encryption time between the two kinds of results

From the results in Figure 3, we can see that with the gradual increase of the key space, the encryption time of the BGV algorithm increases gradually from 0.2 seconds, and the encryption time increases rapidly. When the key space is increased to 700MB, the encryption time has reached 1.38 seconds.

For the cloud storage data based on homomorphic encryption algorithm encryption method, with the increasing of the key space, its encryption time is also increased, but the increase amplitude was significantly less than the BGV algorithm. When the key space is increased to 700MB, the encryption time is also less than 0.7 seconds.

5 Conclusions

A new storage framework based on the idea of the encryption algorithm is designed. A comparison experiment with BGV algorithm is performed; experimental results show that the proposed method for cloud storage data retrieval based on homomorphic encryption algorithm in data encryption, data decryption, data retrieval has a better performance.

References

1. Duncan, B., Whittington, M. : Reflecting on Whether Checklists Can Tick the Box for Cloud Security[C]. IEEE International Conference on Cloudcom, 805-810. (2014)
2. Singh, J., Pasquier, T., Bacon, J., Ko, H., Eyers., D. : 20 Cloud Security Considerations for Supporting the Internet of Things [J]. Internet of Things Journal of IEEE Transaction, 28,1-16. (2015)
3. Duncan, B., Whittington, M.: Enhancing Cloud Security and Privacy: Broadening the Service Level Agreement [C]. The 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom-15), At Helsinki, Finland,1088-1093. (2015)

4. Sari, A. : A Review of Anomaly Detection Systems in Cloud Networks and Survey of Cloud Security Measures in Cloud Storage Applications[J]. Journal of Information Security, 6(2),142-154. (2015)
5. GJW Kathrine, AO Joseph, R Vijayan. Cloud Security Mechanisms for Data Protection: A Survey[J]. International Journal of Multimedia & Ubiquito, 9(9),81-90. (2014)
6. Yadav, S., Kalaskar. K. : Recent Advances in Cloud Security[J]. Journal of Computers, 5(10), 2156-2163. (2014)
7. Kumar, SN. : A Survey on Secure Cloud: Security and Privacy in Cloud Computing[J]. American Journal of Systems and Software, 4(1), 14-26. (2016)
8. Ryoo, J., Rizvi, S., Aiken, W., Kissell, J. : Cloud Security Auditing: Challenges and Emerging Approaches[J]. IEEE Security & Privacy Magazine, 12(6),68-74. (2014)
9. Sharma, S., Gupta, G., Laxmi, PR. : A Survey on Cloud Security Issues and Techniques[J]. Computer Science, 4(1),111-118. (2014)
10. Martini, B., Do, Q., Choo, KKR. : Conceptual evidence collection and analysis methodology for Android devices Cloud Security Ecosystem[J]. Cloud Security Ecosystem, 285-307. (2016)

