ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 5, Special Issue 11, May 2016

International Conference on Innovations in Engineering and Technology (ICIET'16)

6th & 7th April 2016

Organized by

K.L.N. College of Engineering and Technology, Madurai, Tamil Nadu, India

Significant Data Retrieval with Adequate Privacy in Cloud Service Providers

Prem Paul. G, Saravanakumar. T, Santhosh Kumar.T, Venkatesh.C

Department of Computer Science Engineering, K.L.N. College of Engineering, Pottapalayam, Sivagangai India

ABSTRACT: The huge amount of data at data owners outsource into the cloud in a encrypted for the purpose of security. The large volume of data in data centers has experienced a dramatic growth. Therefore it is essential to develop efficient and reliable cipher text search techniques. This will make it even more challenging to design cipher text search schemes that can provide efficient and reliable online information retrieval on large volume of encrypted data. In this paper, data ciphering algorithm is proposed to improve security within a big data environment. The proposed algorithm increases the key ranges of encryption process. Furthermore, the proposed method has an advantage over the traditional method in the security of documents and relevance of retrieved documents.

KEYWORDS: Cloud computing, ciphertext search, encryption algorithm, ranked search, multi-keyword search, big data and security.

I. INTRODUCTION

As we step into the big data era, large amount of data are produced world-wide per day. Enterprises and users who own a large amount of data usually choose to outsource their precious data to cloud facility in order to reduce data management cost and storage facility spending. As a result, data volume in cloud storage facilities is experiencing a dramatic increase. Although cloud server providers (CSPs) claim that their cloud service is armed with strong security measures, security and privacy are major obstacles preventing the wider acceptance of cloud computing service.A traditional way to reduce information leakage is data encryption. However, this will make server-side data utilization, such as searching on encrypted data, become a very challenging task. In the recent years, researchers have proposed many cipher text search schemes by incorporating the cryptography techniques. This method does not provide enough security to protect the user data. Thus here a data encryption algorithm is proposed to improve the data security. This algorithm improves security by increasing the key range of data encryption. It proves encrypted data is unbreakable without knowing the security key. A hierarchical clustering is used to cluster the documents based on the minimum relevance threshold, and then partitions the resulting clusters into sub-clusters until the constraint on the maximum size of cluster is reached. Comparing with all the documents in the dataset, the number of documents which user aims at is very small. Due to the small number of the desired documents, a specific category can be further divided into several sub-categories. The keyword entered at the client side is also encrypted to provide keyword privacy. These encrypted keywords are searched over the cloud storage. Cloud server

will first search the categories and get the minimum desired
sub-category. Then the cloud server will select the desired k documents from the minimum desired sub-category. The value of k is previously decided by the user and sent to the cloud server. The retrieved documents are ranked and sent to the client.

II. EXISTING SYSTEM

The searchable encryption which provides text search function based on data encryption has been widely used, especially in preserving privacy and improving efficiency are the recent year strategy.

The existing system uses the data encryption algorithm to encrypt the user data and the indexes. These encrypted data and their indexes outsourced into cloud server. These data along with their indexes are clustered and then stored in the cloud. But these encrypted data are not completely secure since the length of the secret key is not enough to maintain privacy.

To improve search strategies, a number of conjunctive keyword search methods are used. These methods show large overhead, such as communication cost by sharing secret or computational cost by bilinear map. Due to the lack of the security analysis for frequency information and practical search performance, it is unclear whether their scheme is secure and efficient or not. Earlier approach represents a novel architecture to solve the problem of multi-keyword ranked search over encrypted cloud data. But the search time of this method grows exponentially accompanying with the exponentially increasing size of the document collections.
At the stage of index building process, the relevance between documents is ignored. As a result, the relevance of plaintexts is concealed by the encryption, users expectation cannot be fulfilled well. For example: given a query containing Mobile and Phone, only the documents containing both of the keywords will be retrieved by traditional methods. But if taking the semantic relationship between the documents into consideration, the documents containing Cell and Phone should also be retrieved.

Keywords searched by the users are also encrypted for the purpose of keyword privacy. But the encryption levels of these keywords are not enough to provide keyword privacy.

III. PROPOSED SYSTEM

In this paper, a multi-keyword ranked search over encrypted data in cloud servers with adequate privacy to maintain the privacy of user data is proposed(refer fig 1.1). In order to enhance the search efficiency, relationships between different documents over the encrypted domain are maintained. In the proposed architecture, the search time has a linear growth accompanying with an exponential growing size of data collection. To speed up the searching process relevance score between the query and documents are computed. So only the documents which are specific to user queries are the results of the search. The proposed system also increases the key length of the encryption algorithm to improve the security and privacy.This system provides adequate enough security to the data and indexes. When these data along with indexes are outsourced the hierarchical clustering method is used to store the data sets. The proposed algorithm supports variable length key size. So that it is flexible in encrypting data with varied key sizes.

According to the proposed clustering method, every document will be dynamically classified into a specific cluster which has a constraint on the minimum relevance score between different documents in the dataset. The relevance score is a metric used to evaluate the relationship between different documents. Due to the new documents added to a cluster, the constraint on the cluster may be broken. If one of the new documents breaks the constraint, a new cluster center will be added and the current document will be chosen as a temporal cluster center. Then all the documents will be reassigned and all the cluster centers will be reelected. Therefore, the number of clusters depends on the number of documents in the dataset and the close relationship between different plain documents. In other words, the cluster centers are created dynamically and the number of clusters is decided by the property of the dataset. According to the proposed method, the number of clusters and the minimum relevance score increase with the increase of the levels whereas the maximum size of a cluster reduces. Depending on the needs of the grain level, the maximum size of a cluster is set at each level. Every cluster needs to satisfy the constraints. If there is a cluster whose size exceeds the limitation, this cluster will be divided into several sub clusters.

In the search phase, the cloud server will first compute the relevance score between query and cluster centers of the first level and then chooses the nearest cluster. This process will be iterated to get the nearest child cluster until the smallest cluster has been found. The cloud server computes the relevance score between query and documents included in the smallest cluster. cloud server will trace back to the parent cluster of the smallest cluster and the brother clusters of the smallest cluster will be searched. This process will be iterated until the number of desired documents is satisfied or the root is reached. Due to the special search procedures, the rankings of documents among their search results are different with the rankings derived from traditional sequence search. Therefore, the rank privacy is enhanced.

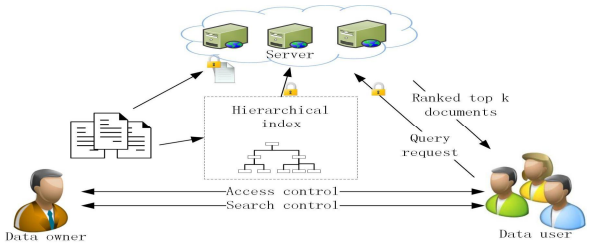

Figure 1: Architecture of encrypted data retrieval

The above architecture consists of three modules.

The first module is carried out between the data owners and the cloud servers. The encryption takes place with increased key size and it lies between 32 to 448 bits. The encrypted data along with its indexes are uploaded into the cloud servers.

The next module involves storing the encrypted data ad its indexes into the cloud server. The hierarchical clustering method is used to cluster the data based on the class labels so that the data with similar class labels belong to same cluster. The cluster size increases according to the amount of data with similar class labels. To reduce overhead the clusters are divided into sub clusters. These sub clusters are again divided until a specific constraint is satisfied.

The last module takes place between the cloud servers and the users. The access and search controls for the users should provide by the data owners which involves the request and response model. Then the users have the right to access the document. Once the keywords are entered by the user, these keywords are encrypted and then searched over the encrypted domain in the server. The matched documents are retrieved and then efficiently ranked. The ranking process depends on the number of keywords matching. For example, the document with more number of keyword occurrences has the first priority in ranking. Similarly, the rest of the matched cases are ranked. The ranked results are sent to the user as the result of the search.

IV. GOALS

Privacy Preserving: A number of requirements that are to be focused in preserving the privacy are given.

Index privacy: Index privacy means the ability to frustrate the adversary attempt to steal the information stored in the index. Such information includes keywords and the Term Frequency of keywords in documents, the topic of documents, and so on.

Data privacy: Data privacy preserves the confidentiality and privacy of documents. The external source cannot get the plaintext of documents stored on the cloud server if data privacy is guaranteed. To achieve data privacy the Symmetric cryptography is a conventional way.

Keyword privacy: It is important to protect users query keywords. Secure query generation algorithm should not leak any information about the query keywords.

Rank privacy: Rank order of search results should be well protected. If the rank order remains unchanged, the adversary can compare the rank order of different search results, further identify the search keyword.

Search efficiency: The time complexity of search time of the given scheme needs to be logarithmic against the size of data collection in order to deal with the explosive growth of document size in big data scenario.

Retrieval accuracy: The relevance between the query and the documents in result set is calculated and then the relevance of documents in the result set is found out.

V. ARCHITECTURE AND ALGORITHM

5.1 SYSTEM MODEL

In this section, a ciphertext scheme is introduced. This model is similar as that of MRSE-HCI. The hierarchical index structure is used to overcome the problems in sequence index. Here, every document is indexed by a vector. Every dimension of the vector stands for a keyword and the value represents whether the keyword appears or not in the document. Similarly, the query is also represented by a vector.

Due to the fact that all the documents outsourced to the cloud server are encrypted, the semantic relationship between plain documents over the encrypted documents is lost. In order to maintain the semantic relationship between plain documents over the encrypted documents, a clustering method is used to cluster the documents by clustering their related index vectors.

After creating the index for the data, the data privacy mechanisms in cloud servers is not enough to maintain the security. To avoid the information leakage the documents are ciphered with more secret key size. This process of encryption involves encrypting data with large key size so that information leakages in cloud servers are minimized.

With the dramatic growth of data volume in the data center makes the conventional sequence search approach inefficient.
To improve the search efficiency, a hierarchical clustering method is used. This hierarchical approach clusters the documents based on the minimum relevance threshold at different levels, and then partitions the resulting clusters into sub-clusters until the constraint on the maximum size of cluster is reached.

In the search phase, cloud server calculates the relevance score between the query and documents by computing the inner product of the query vector and document vectors and returns the target documents to user according to the top k relevance score.

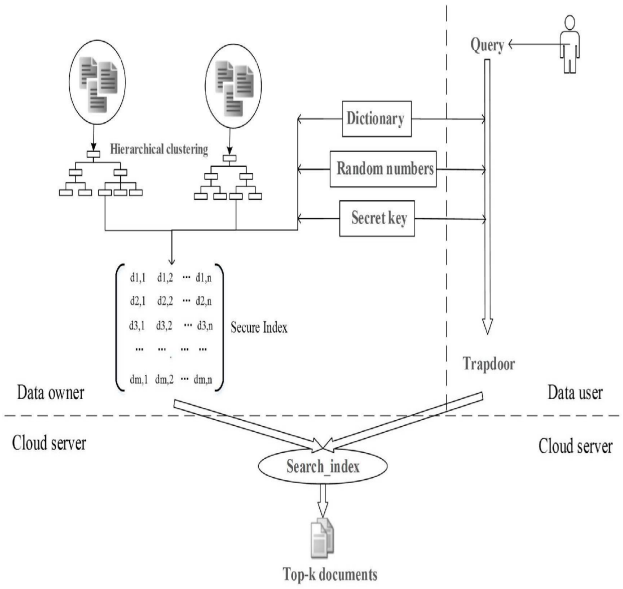

Figure 2: Cipher text search architecture

5.2 CIPHERTEXT ARCHTECTURE

The proposed encryption algorithmis a symmetric block cipher that can be effectively used for encryption andsafeguarding of data. It is used in two modules of the main architecture. In Module 1 it is used to encrypt data and indexes and then in Module 3 the user keywords are also encrypted by this algorithm.

This algorithm has a scalable key, from 32 bits to at least 256 bits. And ituses simple operations that are efficient on microprocessors. Itemploys precomputable subkeys.

The proposed algorithm uses a large number of subkeys. These keys must be pre computed before anydata encryption or decryption.Thus here both encryption block and decryption blocks are explained.

5.2.1 ENCRYPTION

The blowfish algorithm is a Feistel Network, iterating a simple encryption function 16 times. The block size is 64 bits, and the key can be any length up to 448 bits. Although there is a complex initialization phase required before any encryption can take place, the actual encryption of data is very efficient on large microprocessors.

The Feistel network is a general method of transforming any function (usually called an Ffunction) into a permutation. It is used in our block cipher designs. The working of a Feistal Network is described as it split each block into halves. Right half becomes new left half. New right half is the final result when the left half is XOR'd with the result ofapplying *f* to the right half and the key.

The Feistel network is divided into two parts:

Key-expansion: It will convert a key of at most 448 bits into several subkey arrays totaling 4168 bytes.

Data encryption: It is having a function to iterate 16 times of network. Each round consists of a key-dependent permutation, and a key- and data- dependent substitution. All operations are XORs and additions on 32-bit words. The only additional operations are four indexed array data lookup tables for each round.

FUNCTION F

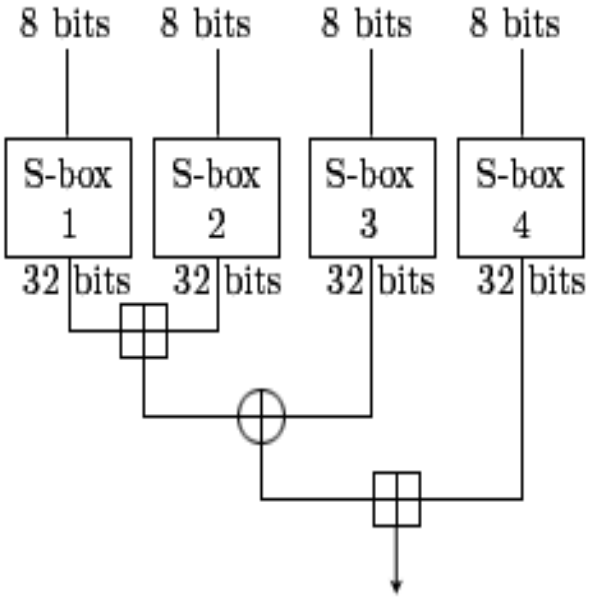

Figure 3: Block diagram for F function

The equation for F function is given below
$F = ((S1[a] + S2[b] \bmod 2^{32}) \text{ XOR } S3[c]) + S[d] \bmod 2^{32})$
Blowfish has 16 rounds. Each round consists of a key dependent permutation, and a key- and data-dependent substitution. All operations are XORs and additions on 32-bit words. The only additional operations are four indexed array data lookups per round.

1. Sub keys Generation: Blowfish uses a large number of sub keys. Thesekeys must be pre computed before any data encryption ordecryption. The P-array consists of 18 32-bit subkeys: P1,P2,..., P18. There are also four 32-bit S-boxes with 256 entrieseach: S1,0, S1,1,..., S1,255; S2,0, S2,1,..,, S2,255; S3,0, S3,1,...S3,255; S4,0, S4,1,..,, S4,255.

2. Encryption Process: Blowfish has 16 rounds. Theinput is a 64-bit data element, x.
Divide x into two 32-bithalves: xL, xR.
Then, for i = 1 to 16:
Xl = xL XOR Pi
xR = F(xL) XOR xR
And Swap xL and xR

After the sixteenth round, swap xL and xR again to undo thelast swap. Then, xR = xR XOR P17 and xL = xL XOR P18.
Finally, recombine xL and xR to get the cipher text.

Function F looks like this: Divide xL into four eight-bitquarters: a, b, c, and d.
Then, F(xL) = ((S1,a + S2,b mod 232)XOR S3,c) + S4,d mod 232.

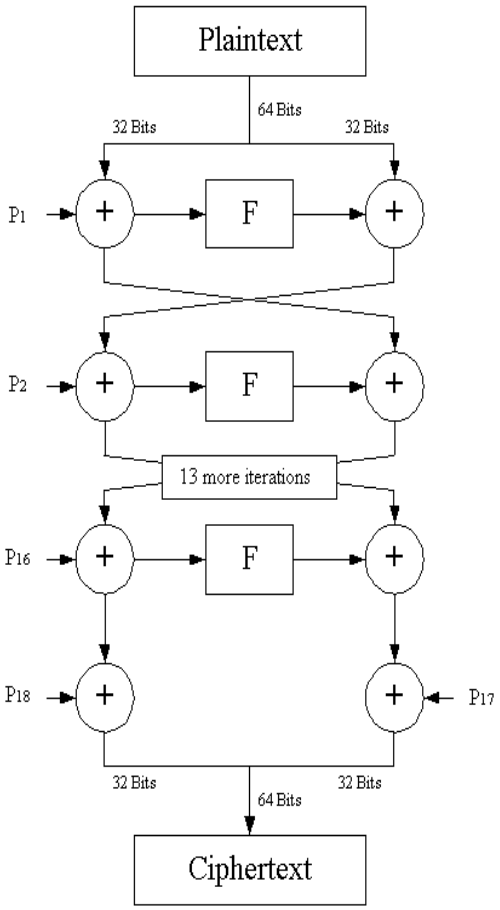

Figure 4: Conversion of plain text to cipher text

GENERATING THE SUBKEYS

The sub keys are calculated using the Blowfishalgorithm:

1. Initialize first the P-array and then the four S-boxes, in order, with a fixed string.
This string consists of the hexadecimal digits of pi (less the initial 3):
P1 = 0x243f6a88, P2 = 0x85a308d3,
P3 = 0x13198a2e, P4 = 0x03707344, etc.

2.XOR P1 with the first 32 bits of the key, XOR P2 with the second 32-bits of thekey, and so on for all bits of the key (possibly up to P14). Repeatedly cyclethrough the key bits until the entire P-array has been XORed with key bits. (Forevery short key, there is at least one equivalent longer key; for example, if A is a64-bit key, then AA, AAA, etc., are equivalent keys.)

3. Encrypt the all-zero string with the Blowfish algorithm, using the sub keys described in steps (1) and (2).

4. Replace P1 and P2 with the output of step (3).

5. Encrypt the output of step (3) using the Blowfish algorithm with the modified sub keys.

6. Replace P3 and P4 with the output of step (5).

7. Continue the process, replacing all entries of the P array, and then all four S-boxes inorder, with the output of the continuously changing Blowfishalgorithm.

In total, 521 iterations are required to generate all required subkeys. Applications canstore the subkeys rather than execute this derivation process multiple times.

5.2.2 DECRYPTION:

The decryption process described here is the reverse process of the encryption discussed above, except that P1, P2...

P18 are used in thereverse order.

This decryption algorithm is identical to the encryption algorithm step by step in the same order, only with the sub-keys applied in the reverse order

The figure 5 represents the process of converting the cipher text to plain text.

Both the data owner's data and the keywords entered by the user at client side are encrypted using

Decryption is quite simple and accomplish by merely inverting the P17 and P18 cipher blocks and using P entries in reverse. The S-boxes and P-boxes are initialized with values from hex digits of pi. The variable length user-input key is then XOR with P-entries.

Then a block of zeros is encrypted, and this result is used for P1 and P2 entries. The cipher text resulting from the encryption of a zero block is then encrypted again and use for P3 and P4. This process continues until every P-box entry and S-box entry has been replaced, resulting in 521 successive key generations. This involves about 4KB of data processing. This relatively complex key schedule makes Blowfish an effective and durable cryptographic algorithm.

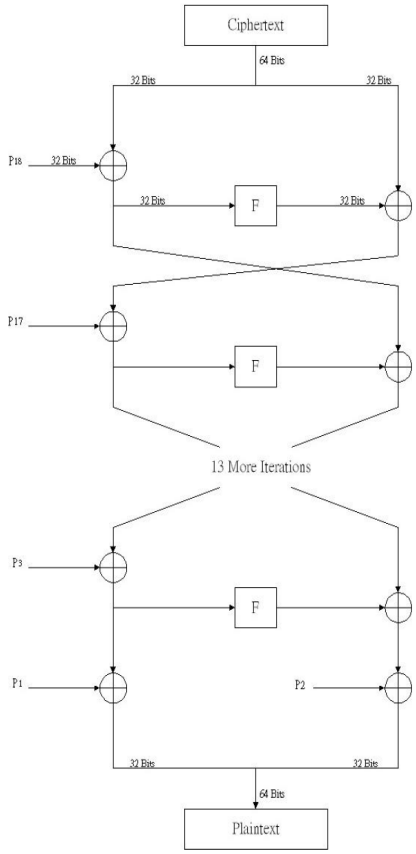

Figure 5: Conversion of cipher text to plain text

VI. HIERARCHICAL CLUSTERING ALGORITHM

A number of hierarchical clustering methods are available. But these methods are not efficient as that of partition clustering method in term of time complexity. K-means and K-medoids are popular partition clustering algorithms. But the k is fixed in the above two methods, which cannot be applied to the situation of dynamic number of cluster centers. Here a quality hierarchical clustering (QHC) algorithm is used. It is based on the novel dynamic K-means.

The QHC algorithm is illustrated in the Fig.5. It goes like that. Every cluster will be checked on whether its size exceeds the maximum number TH or not. If the answer is "yes", this"big" cluster will be split into child clusters which are formed by using the dynamic K-means on the documents of this cluster.

This procedure will be iterated until all clusters meet the requirement of maximum cluster size. Clustering procedure is illustrated in Fig.6. All the documents are denoted as points in a coordinate system. These points are initially partitioned into two clusters by using dynamic K-means algorithm when the k = 2. Then these two clusters are checked

to see whether their points satisfy the distance constraint.

The second cluster does not meet this requirement, thus a new cluster center is added with k = 3 and the dynamic k-means algorithm runs again to partition the second cluster into two parts. Then the data owner checks whether these clusters size exceed the maximum number TH. Cluster 1 is split into two sub-clusters again due to its big size

Algorithm *Quality Hierarchical Clustering (QHC)*

1, input documents and set the size threshold *TH*
2, build cluster set C_0 in first level by *dynamic k-means*
3, **while** there are new cluster set C_i
4, **for** every cluster $C_{i,i}$
5, **if** the size of $C_{i,i}$ is bigger than *TH*
6, split this cluster into sub-clusters C_{i+1}
7, **Until** all clusters match the size constraint

Figure 6: Algorithm for Quality Hierarchical Clustering

VII. SEARCH ALGORITHM

The cloud server needs to find the cluster that most matches the query. With the help of cluster index I_c and document classification DC, the cloud server uses an iterative procedure to find the best matched cluster. Following instance demonstrates how to get matched one:

1) The cloud server computes the relevance score between Query T_w and encrypted vectors of the first level cluster centers in cluster index I_c, then chooses the i^{th} cluster center $I_{c;1;i}$ which has the highest score.

2) The cloud server gets the child cluster centers of the cluster center, then computes the relevance score between T_w and every encrypted vectors of child cluster centers, and finally gets the cluster center $I_{c;2;i}$ with the highest score. This procedure will be iterated until that the ultimate cluster center $I_{c;l;j}$ in last level l is achieved.

VIII. CONCLUSION

In this paper, cipher text search in the area of cloud storage is discussed. Here the problem of maintaining the semantic relationship between different plain documents over the related encrypted documents and give the design method to enhance the performance of the semantic search is explored. Also the security mechanism in the cloud was improved by increasing the key range of the encryption algorithm. This paper also resolves the security problem and search efficiency in the previously proposed system. This paper also expresses the method for the keyword privacy, rank privacy, and index privacy.

REFERENCES

[1] D. X. D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proc. S & P, BERKELEY, CA, 2000, pp. 44-55.
[2] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. EURO-CRYPT, Interlaken, SWITZERLAND, 2004, pp. 506-522.
[3] Y. C. Chang, and M. Mitzenmacher, "Privacy preserving key-word searches on remote encrypted data," in Proc. ACNS, Columbia Univ, New York, NY, 2005, pp. 442-455.
[4] H. Pang, and K. Mouratidis, Authenticating the query results of text search engines, Proc. VLDB Endow., vol. 1, no. 1, pp. 126-137, Aug. 2008.
[5] Y. H. Hwang, and P. J. Lee, "Public key encryption with conjunctive keyword search and its extension to a multi-user system," in Proc. Pairing, Tokyo, JAPAN, 2007, pp. 2-22.
[6] L. Ballard, S. Kamara, and F. Monrose, "Achieving efficient conjunctive keyword searches over encrypted data," in Proc. ICICS, Beijing, China, 2005, pp. 414-426.
[7] D. Boneh, G. Di Crescenzo, R. Ostrovsky, et al. Public key encryption with keyword search[C].Advances in Cryptology- Eurocrypt 2004. Springer Berlin Heidelberg, 2004: 506-522.
[8] D. Cash, S. Jarecki, C. Jutla, et al. Highly-scalable searchable symmetric encryption with support for boolean queries[M].Advances in CryptologyCRYPTO 2013. Springer Berlin Heidelberg, 2013: 353-373.
[9] S. Kamara, C. Papamanthou, T.Roeder. Dynamic searchable symmetric encryption[C].Proceedings of the 2012 ACM conference on Computer and communications security. ACM, 2012: 965-976.
[10] Curtmola R, Garay J, Kamara S, et al. Searchable symmetric encryption: improved definitions and efficient constructions[C]//Proceedings of the 13th ACM conference on Computer and communications security. ACM, 2006: 79-88.
[11] Chase, M., and Kamara, S. (2010). Structured encryption and controlled disclosure. In Advances in Cryptology-ASIACRYPT 2010 (pp. 577-594). Springer Berlin Heidelberg
[12] S. C. Yu, C. Wang, K. Ren, and W. J. Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing,"
in Proc. IEEE INFOCOM, San Diego, CA, 2010, pp. 1-9.
[13] I. H. Witten, A. Moffat, and T. C. Bell, Managing gigabytes: compressing and indexing documents and images, 2nd ed., San Francisco: Morgan Kaufmann, 1999.
[14] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in Proc. Berkeley Symp. Math. Stat. Prob, California, USA, 1967, p. 14.