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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

Secure Multi-Keyword Rank Search on Encrypted Cloud Data for Multiple Users

Deepali D. Rane [1], Dr. V. R. Ghorpade [2]

Assistant Professor, Department of Information Technology, D. Y. Patil College of Engineering, Akurdi, Pune, India[1]

Professor, Department of Computer Science, D. Y. Patil College of Engineering & Technology, Kolhapur, Maharashtra, India [2]

ABSTRACT: Public cloud is now a rapid growing trend for storing user's data. Most of the users now a day's placing their personal and professional data on the public cloud. As they outsource their confidential data on cloud. They lose physical possession of their private data. Public cloud is one way to store user's data but it is not trusted storage service. In this paper, two of the privacy preserving issues about accessing the cloud data has been identified i.e. acuteness of keywords sent in queries and the data fetched as a result of those queries. Both of these should be hidden from intruders. To keep the documents private, they should get encrypt before outsourcing to the cloud. Privacy of the data has been achieved using symmetric key cryptography algorithm, we have used Twofish. To fetch the documents of user interest, user should fire a query consisting of multiple keywords which will in turn return the top k ranked documents. As we don't want to disclose neither keyword from query nor query pattern, we have developed fully privacy preserving system by encrypting search pattern as well as secret key. Indexing has been developed to build an index of keywords from documents. Index will be used to retrieve documents in response to search query by using the principle of keyword matching. This paper has analysed and implemented Lucene indexing algorithm. Ranking of the results has been developed so as to improve the search result correctness as well as to embellish the user searching experience**.**

KEYWORDS: Privacy Preserving, Sensitivity of Keywords, Twofish, Multi-Keyword query, Ranking, Indexing, Lucene

I. INTRODUCTION

Cloud computing means storing and accessing data and programs over the Internet instead of your computer's hard drive. The cloud is just a metaphor for the Internet. There are three types of cloud: public cloud, private cloud and hybrid cloud. In today's era many users want to have a secure storage for their data, it this situation cloud comes into picture. Users are motivated to outsource their personal as well as professional data to public cloud, but they want their data to be safe at cloud side because as they outsource it, they lose physical possession of their data. Public cloud storage is not trusted storage. To protect data confidentiality and unauthorized access to the cloud data, owners are motivated to encrypt the data before outsourcing to the cloud. As user is going to store encrypted data at cloud side, traditional searching will not be effective. To meet the effective searching on the encrypted cloud data, Multi-Keyword query should be formed and fired so as to get the top relevant data of user interest. Numbers of public cloud service providers are there like Amazon EC2, Google drive, Rackspace, Jelastic etc which will prove users to freely store their data on cloud.

II. RELATED WORK

Previous systems have implemented various searching schemes which have utilized bilinear maps and also which are based on public key encryption technique. When searching for a particular document, either single keyword searching or single user searching has been developed. Users are sending the query consisting of keywords, hence

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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

unable to hide the search pattern.[1] Some schemes are developed in which user must have knowledge about all the valid keywords and their respective positions as mandatory information so as to generate a query [2].

Disadvantages of Existing System can be ,

1. Single-keyword search without ranking
2. Boolean keyword searching without ranking
3. Single-keyword search with ranking
4. Rarely sorting of the results i.e. no index creation and ranking
5. Single User search

III. PROPOSED SYSTEM

The primary goal of the proposed system is not only to fetch the top priority data in response to users query but also to embellish the user searching experience. This requirement makes it mandatory to develop a rank based system which will support multi-keyword searching, as single keyword searching will yield too much unnecessary traffic.

A. PROBLEM FORMULATION

As public cloud enables users to outsource their personal as well as professional data to cloud, whenever and wherever necessary they should fetch their data by firing the query, which itself consist of multiple keywords. To improve the search results on cloud data, system should construct an index of keywords and should retrieve only top *k* relevant documents matching the keywords specified in the search query.

B. DESIGN GOALS

1. To protect data privacy by encrypting documents before outsourcing
2. Rank based retrieval of the documents.
3. To easily access the encrypted data by multi keyword rank search using keyword index.

C. SYSTEM ARCHITECTURE

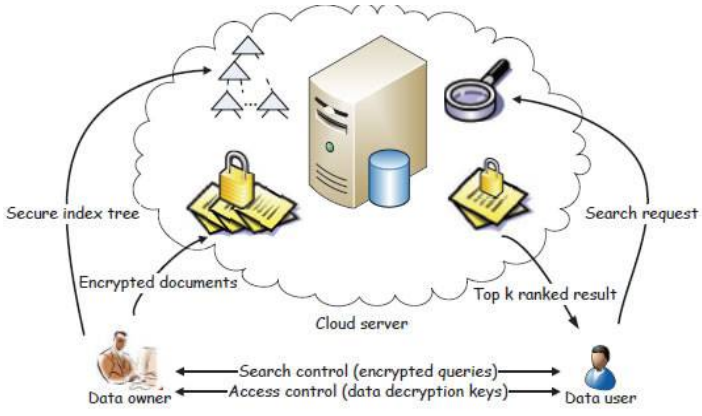

Figure.1 System Architecture

Proposed system will solve the dilemma of searching over encrypted cloud data by firing multi-keyword search query with conserving the system-wide privacy in cloud computing. Finding Keyword relevance is an intermediate similarity measure that has been chosen among various semantics, which uses number of keywords from the search query. Keywords appearing in the document will be used to quantify the relevance of that document to the query. Ranking subsystem has been developed. It proves to be very effective in implementing and returning the documents which are highly relevant to terms submitted in search query. So as to protect data privacy and to prevent accesses from unauthorized user, sensitive data must get encrypted before getting outsourced to the cloud. Rank based search

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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

can appropriately remove unnecessary network traffic by sending back only the top relevant data of user's interest. This system can improve the searching results as well as enhances the user searching and fetching experience by developing ranking system which will support multiple keywords in search query. Single Keyword search query will yield too much unnecessary results so there is necessity of supporting multi-keyword query.

D. FRAMEWORK

Ranked searchable scheme consists of four steps (Encryption- Decryption of document, Index Construction, Query Generation, and Ranked searching).

Cloud server will generate public key - private key pair

1. Cloud server has private key. Private Key will be used for RSA decryption
2. Each Cloud user will have public key. Public key is used for RSA encryption
3. User wants to store a document on cloud.
4. First he will encrypt secret key (e.g. password). Then he will upload the document and encrypted secret key
5. Server will generate index for the new document
6. After creating index, secret key will get decrypted by server using RSA with private key.
7. Then server will encrypt the document with decrypted secret key using Twofish algorithm. Discard the decrypted secret key and original document
8. The Encrypted document, secret key and index will get stored on cloud server.
9. Now user wants to retrieve the document
10. User will give a search query; this query will get encrypted using user's public key and sent to server for search
11. Search query will decrypt by server and searched in index
12. Ranked results will get displayed to user
13. User will select a document d1, and then server will ask for secret key
14. If the secret key matches with key stored on server the user will get granted with the access to document and decrypted document will returned in response

IV. METHODOLOGY

A. ENCRYPTION DECRYPTION OF DOCUMENT

Before outsourcing to the cloud, each document has to be encrypted .For encryption purpose, Twofish algorithm has been implemented. Workflow of Twofish includes splitting 128 bit plaintext into 4 words each of 32bits. Whitening is the second phase which is followed by G function which includes inputting word X and decomposing into 4 bytes. Each byte will perform in its own key dependent S Box. It accepts 8 bits of input, and outputs 8 bits of result. These 4 outputs will be considered as a vector of length four and then multiplied by MDS matrix Next step is Pseudo-Hadamard Transform (PHT) is a simple mixing operation. 32-bit PHT is used to mix the outputs from two parallel 32-bit G functions. And at last cipher Text will get produced.

Table.1 Performance comparison based on safety factor [6]

Twofish	2.67
AES	1.11/1.33/1.56
18-round AES	2.00
24-round AES	2.67

B. KEYWORD INDEXING

Lucene incremental algorithm has following steps:

- Keep stack of segment indices.
- For each incoming document, build index.
- Push new indexes on stack.

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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

- Let A=5 be the merge factor, M=∞

```
For (size=1; size<M; size*=A)
{
    If (there are A indexes with size docs on top of stack)
        {
            Pop them off the stack;
            Merge them into a single stack;
            Push the merged index onto the stack;
        }
    Else
        {
            Break;
        }
}
```

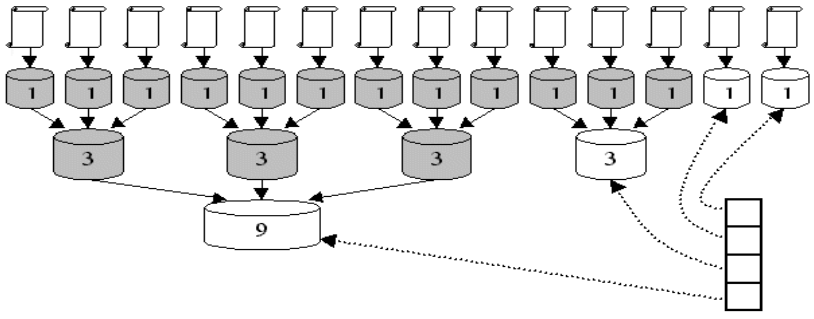

Figure.2 Working of Lucene Indexer

Above figure shows detailed working of Lucene. As the document will get uploads index segment will get created. According to merge factor, index segments will get merged. Index contains hashed values of keywords

C. RSA IMPLEMENTATION FOR ENCRYPTION OF SECRET KEY & SEARCH QUERY AS WELL AS CHECKING RANK OF DOCUMENTS.

1. RSA implementation

RSA is one of the first practical public-key cryptosystems and is widely used for secure data transmission. In such a cryptosystem, the encryption key is public and differs from the decryption key which is kept secret. In proposed system, secret key will get encrypted using RSA's public key and send to cloud and at server side it will get decrypted to encrypt plain text. Using this secret key and search query will not get violated by intruders. To use RSA's public key it should be available to every user. We have developed a mechanism to make public key available to every data owner. Those who wants RSA public key they have to select? Option and then enter email address in which they want to receive public key file. After clicking on send me public key file, system will send public key file to the appropriate email address.

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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

2. Checking rank of document

In response to users query cloud server returns a bunch of documents called Hit list. has been made so as to check whether server has returned the documents in priority order or not. **Keyword Relevance** is the main principle chosen behind ranking the documents. Document having greater score will appear first than others having low score.

i. Scoring

Scoring uses a combination of the Vector Space Model (VSM) of Information Retrieval and the Boolean model to determine how relevant a given Document is to a User's query. In general, the idea behind the VSM is the more times a query term appears in a document relative to the number of times the term appears in all the documents in the collection, the more relevant that document is to the query. It uses the Boolean model to first narrow down the documents that need to be scored based on the use of Boolean logic in the Query specification.

The factors involved in scoring are as follows:

a) tf = term frequency in document = measure of how often a term appears in the document
b) idf = inverse document frequency = measure of how often the term appears across the index
c) coord = number of terms in the query that were found in the document
d) lengthNorm = measure of the importance of a termaccording to the total number of terms in the field
e) queryNorm = normalization factor so that queries can be compared
f) Boost (index) = boost of the field at index-time
g) Boost (query) = boost of the field at query-time

ii. Scoring Algorithm

In the typical search application, a Query is passed to the Searcher, beginning the scoring process. Once inside the Searcher, a Collector is used for the scoring and sorting of the search results. These important objects are involved in a search:

a) The Weight object of the Query. The Weight object is an internal representation of the Query that allows the Query to be reused by the Searcher.
b) The Searcher that initiated the call.
c) A Filter for limiting the result set. Note, the Filter may be null.
d) A Sort object for specifying how to sort the results if the standard score based sort method is not desired.

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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

V. RESULT ANALYSIS

The implementation of the Encryption & Decryption, Secure index construction is successfully completed with desirable performance. After index construction it will get compressed and will be stored in .cfs file format. After firing single-keyword query, user will get all documents that contain the specified keyword. Timing function has been added which will calculate encryption & indexing time.

Sr.No	FileSize(kb)	Twofish	Blowfish
1	147	6	48
2	322	10	65
3	741	21	101
4	1548	55	208

Table a) Encryption Time comparison for .txt file.

Sr.No	FileSize(kb)	Twofish	Blowfish
1	174	11	35
2	240	15	42
3	447	25	52
4	854	36	85

Table b) Encryption Time comparison for .doc file.

Sr.No	FileSize(kb)	Twofish	Blowfish
1	159	8	21
2	308	13	32
3	466	23	52
4	1323	34	149

Table c) Encryption Time comparison for .pdf file

Sr.No	FileSize(kb)	Twofish	Blowfish
1	256	9	41
2	486	13	67
3	871	23	84
4	1972	67	220

Table d) Encryption Time comparison for .ppt file.

It is very important to calculate the throughput of an encryption algorithm to know the performance of the algorithm. The encryption time for the algorithm was used to calculate the throughput of an encryption scheme. It indicates the speed of the encryption. The throughput is given by
Throughput= T_P E_t, Where $\mathbf{T_p}$ is the total plaintext and is $\mathbf{E_t}$ the total encryption time.

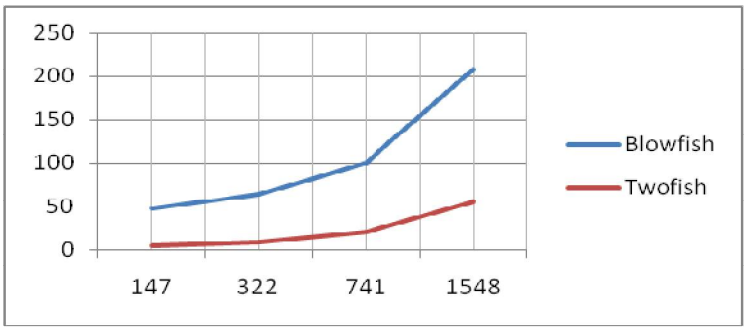

Chart 1: Text file encryption time for Twofish and Blowfish

- For Text files
  T_p =147 +322+ 741 +1548 = 2758
  E_t (Twofish) = 6+10+21+55=92
  E_t (Blowfish) =48 + 65+ 101+208 =422

Throughput for Twofish= T_p/ E_t (Twofish) = 2758/92= **29.98**
Throughput for Blowfish= 2758/422= **6.54**
Above calculations clearly shows that Twofish has higher throughput than Blowfish.

- Encryption time of various File Formats for same size using Twofish

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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

Encryption time has been calculated by subtracting start time from current time. Experiment has been carried out on Jelastic public cloud as well as on local host. When carried out on local host, intel® Core™ i5 CPU with 4GB Ram and 64 bit operating system specification was used.

File Type size=270kb	Encryption Time (milisec)
Text	13
Doc	13
Ppt	13
Pdf	14

Table 2: Comparison of encryption time for same size of file

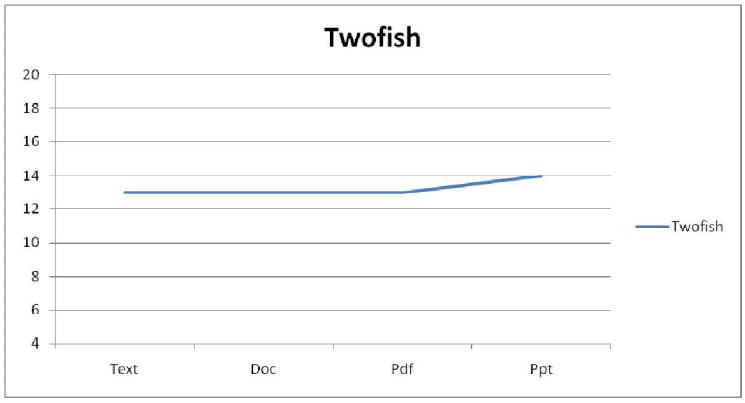

Chart 2: Encryption time comparison for various file types

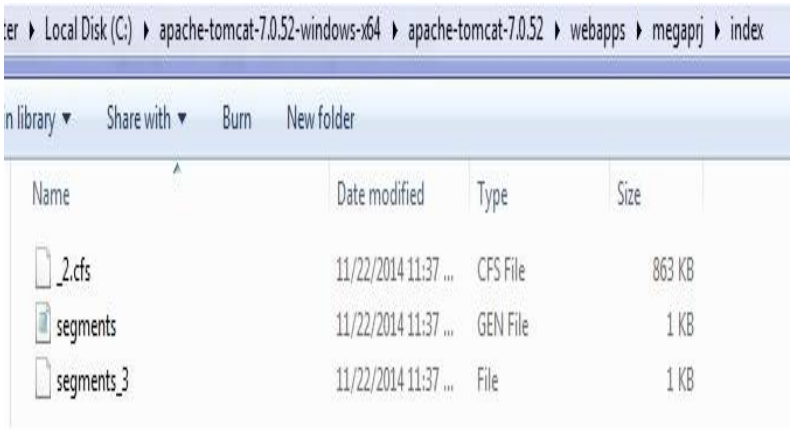

Figure.3 Keyword index on local server

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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

VI. CONCLUSION

We have implemented an automatic text detection technique from an image for Inpainting. Our algorithm successfully detects the text region from the image which consists of mixed text-picture-graphic regions. We have applied our algorithm on many images and found that it successfully detect the text region. Proposed paper has focused on how the user can effectively store their personal documents on cloud while preserving privacy of their documents and whenever and wherever necessary they can retrieve them by sending a query consisting of multiple keywords. Privacy will be achieved by encrypting the queries. In response to the users query, system will match the keywords from query to the documents using "keyword-relevance principle". Top ranked documents will get fetched which will consist of keywords specified by user in query. Checking the rank of the retrieved document can be done by calculating how many docs contains the specified keywords how many times. . In multiple-keyword query, multiple keywords are separated by Boolean keywords like AND OR NOT. System will perform appropriate operations and return results. Boolean expressions like ||, && and! will also be working for searching. To ensure privacy of secret key and search query, both have been encrypted using RSA's public key and at cloud side it will get decrypted for further encryption of documents using Twofish algorithm.

REFERENCES

1. Deepali D. Rane, Dr. V. R. Ghorpade, "Multi-User Multi-Keyword Privacy Preserving Ranked Based Search Over Encrypted Cloud Data", International Conference on Pervasive Computing (ICPC), IEEE 2015.
2. Ning Cao, Cong Wang, Ming Li, KuiRen, Wenjing Lou , "Privacy Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data" IEEE Transactions on parallel and Distributed Systems, Vol. 25,January 2014
3. Ayad Ibrahim, Hai Jin, Ali A.Yassin, DeqingZou, "Secure Rank-Ordered Search of Multi-Keyword Trapdoor over Encrypted Cloud Data" IEEE Asia-Pacific Services Computing Conference 2012
4. Yanjiang Yang, "Towards Multi-User Private Keyword Search for Cloud Computing" IEEE 4th International Conference on Cloud Computing. 2011
5. Qin Liu, Guojun Wag, Jie Wu, "An Efficient Privacy Preserving Keyword Search Scheme in Cloud Computing" IEEE International Conference on Computational Science and Engineering. 2009
6. Twofish : A 128 Bit Block Cipher by Bruce Schneier ,John Kelseyy, Doug Whitingz David, Wagnerx Chris Hall
7. "Analysis of AES and Twofish Encryption Schemes" IEEE Transaction 2011
8. Yong Zhang, Jian-lin Li, "Research and Improvement of Search Engine Based on Lucene" IEEE Transaction 2009
9. QIAN Liping, WANG Lidong, "An Evaluation of Lucene for Keywords Search in Large-scale Short Text Storage" IEEE Transaction 2010
10. Govind S.Pole, Madhuri Potey, " A Highly Efficient Distributed Indexing system based on large cluster of commodity machines" IEEE Transaction 2012