International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 06 | June 2019 www.irjet.net p-ISSN: 2395-0072

Review on Privacy Preserving on Multi Keyword Search over Encrypted Data in Cloud

Ms. Apurva S. Solanke

Student, Department of Computer Science, MSS CET, Jalna, Maharashtra, India

---

Abstract - *As cloud computing becomes widespread, more and more sensitive information are being centralized into the cloud for great convenience and reduced cost in data management. Cloud information proprietors like to outsource records in on encoded form for the capacity of secrecy protecting. User convey data using internet which can be affected by malicious attacks. So to preserve the data privacy of document data should be outsourced in encrypted format which perform traditional data utilization based on plaintext keyword search. As a result allowing an encrypted cloud data search service is of supreme significance. In view of the large number of data users and document in the cloud, it is essential to permit several keywords in the search demand and return document in the order of their appropriate to these keywords. In this proposed system top k query is used to retrieve top matching result from huge encrypted answer set relevant to user interest. In retrieve the document of user interest multi-keyword query has to be fired which will result in accessing the top k ranked document. Lucene indexing technique has been developed to build on index of keywords from document. Index will store hashed values of keyword. Blowfish algorithm is used for encryption and decryption of data.*

Key Words*: Lucene indexing, Multikeyword, Blowfish, top k, Privacy.

1. INTRODUCTION

Cloud computing can be defined as a model for enabling ubiquitous, convenient and on demand network access to shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort from the user side and minimal service provider interaction. Cloud computing is considered the evolution of variety of technologies that come together to change an organizations approach for building their IT infrastructure. Cloud is not simply the latest term for internet though the internet is necessary foundation for the cloud, the cloud something more than internet. Cloud is where you go to use technology when you need it, you need not install anything on your desktop and you do not need to pay for the technology when you are not using it. Cloud computing relies on sharing resources to ensure consistency and economies of scale. Placing our confidential data in hand of cloud storage is critical because of the term data confidentiality has outstanding importance. Our original plain data must be accessible by only trusted parties. Data must be encrypted and must not be accessible by untrusted third parties such as cloud provider, intermediaries and internet. Therefore to ensure data confidentiality and unauthorized access to the data placed on cloud owners have to adopt encryption process on data while placing confidential data on cloud. User will assemble encrypted data on cloud. So, traditional data searching process will not be effective and efficient for encrypted cloud storage. To retrieve efficient result on encrypted data multi-keyword query should be structured and fired. This multi-keyword query helps to find out top related data belonging to user concern. Existing system works on single query searching procedures or on single user with Boolean search procedures. The Existing searching method for data retrieval are only restricted for single keyword queries. These searching method fetch all the related data corresponding to the specified keyword without ranking the data of user interest. Indexing is common operation in web search engines. Large amount of document demands the cloud server instead of returning undifferentiated result only relevant document can retrieve using ranking. Ranking can also eliminate unwanted network traffic for protection purpose the information related to keyword should not be leak while use the ranking scheme.

As we step into the big data era terabyte of data are produced worldwide per data enterprises and users who own 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 dramatic increase.

Search over encrypted data is technique of great interest in the cloud computing because many believe that sensitive data has to be encrypted before outsourcing to the cloud server in order to ensure user data privacy.

2 LITERATURE SURVEY

In some existing methods, for generation of queries user must have to gather all necessary information about all valid keywords along with its positions. Cloud storage acts as database repository as it stores all information and documents of users. One of the cryptographic primitive is searchable encryption which allows private keyword based searching over encrypted database. Existing system has some of the disadvantages like: only single user can search, searching possible for Single or Boolean keyword without ranking, index creation happens very rarely.

Deepali D. Rane et.al, proposed implementation of the encryption and 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. The advantages are protects data privacy by encrypting documents before outsourcing, rank based retrieval of the documents, To easily access the encrypted data by multi keyword rank search using keyword index.

A single keyword searchable encryption schemes usually builds an encrypted searchable index such that, it's content is hidden to the server, unless it is given appropriate trapdoors generated via secret key(s). Early work solves secure ranked keyword search which utilizes keyword frequency to rank results instead of returning undifferentiated results. However, it only supports single keyword search. Where anyone with public key can write to the data stored on server, but only authorized users with private key can search. Traditional single keyword searchable encryption schemes are usually built in a way by creating an encrypted searchable index. Such indexes content will be hidden to the server. The information will be revealed only when the server gives the correct trapdoors that are generated via a secret key(s). The main drawback of single keyword-based search is that it is not comfortable enough to express complex information needs.

2.1 Comparison of Algorithm

Symmetric encryption is also called as secret key encryption which uses a single key for both encryption and decryption processes. Same key is utilized to convey data for encryption process and in decryption process. Conveying key using Internet can be affected by malicious attacks. So if both parties are aware about the keys then symmetric encryption technique is more appropriate for communication to transfer secrete data over internet. Asymmetric encryption is also called as public key private key encryption which utilizes two distinct keys for encryption and decryption processes, public key and the private key respectively. The key can be transformed into an array of bytes for the intention of storing and transferring secretes data. This process is called as wrapping. When key is required for decryption purpose then array of bytes is again converted in key. This process is called as unwrapping. There are various algorithms for data encryption using symmetric and asymmetric encryption keys. Few of them are stated and compared below:

2.2 DES:

DES (Data Encryption Standard) is the very earliest encryption standard developed by NIST (National Institute of Standards and Technology) and commonly used in the commercial, military, and other domains. It was emerged in 1947 by IBM and accepted as a national standard in 1997. DES standard is public & the design criteria used are classified. DES is a 64-bit block cipher which utilizes 56-bit key. This algorithm contains permutation of sixteen rounds block cipher. DES algorithm carried out permutation and substitution procedures on every block of plaintext data which is afterwards XORed with the given input. The same process is carried out 16 times with different sub keys. It is vulnerable to security attacks and secrete data can be easily retrieved.

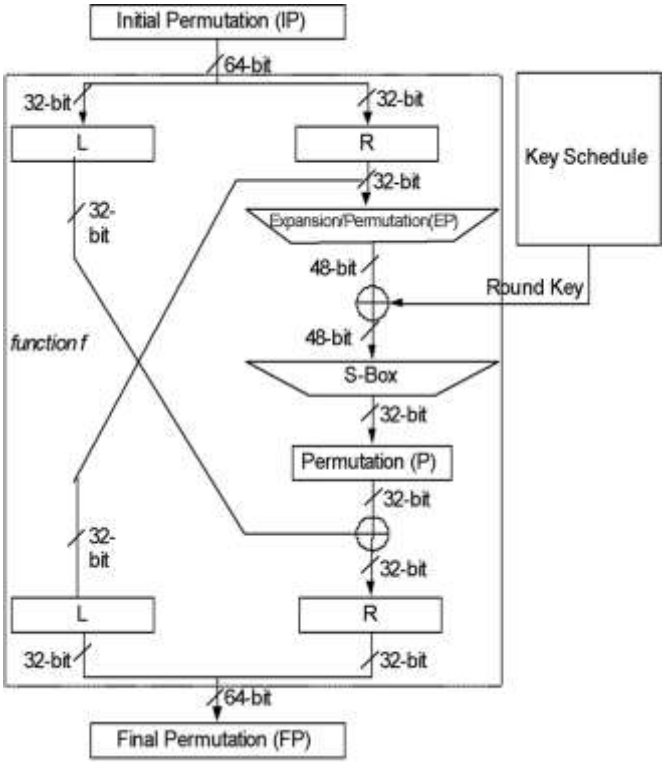


Fig. DES Algorithm

2.3 AES:

AES (Advanced Encryption Standard) was discovered by scientists Joan and Vincent Rijmen in the year 2000. It replaced DES as the official standard of US National Institute for Standards and Technology (NIST). AES works on variable key sizes and variable block sizes of 128, 192 or 256 bits. AES uses Rijndael block cipher and Rijndael key. Due to the number of permutations and combinations in AES contributing to its higher complexity, AES has a higher security as compared to DES002E.

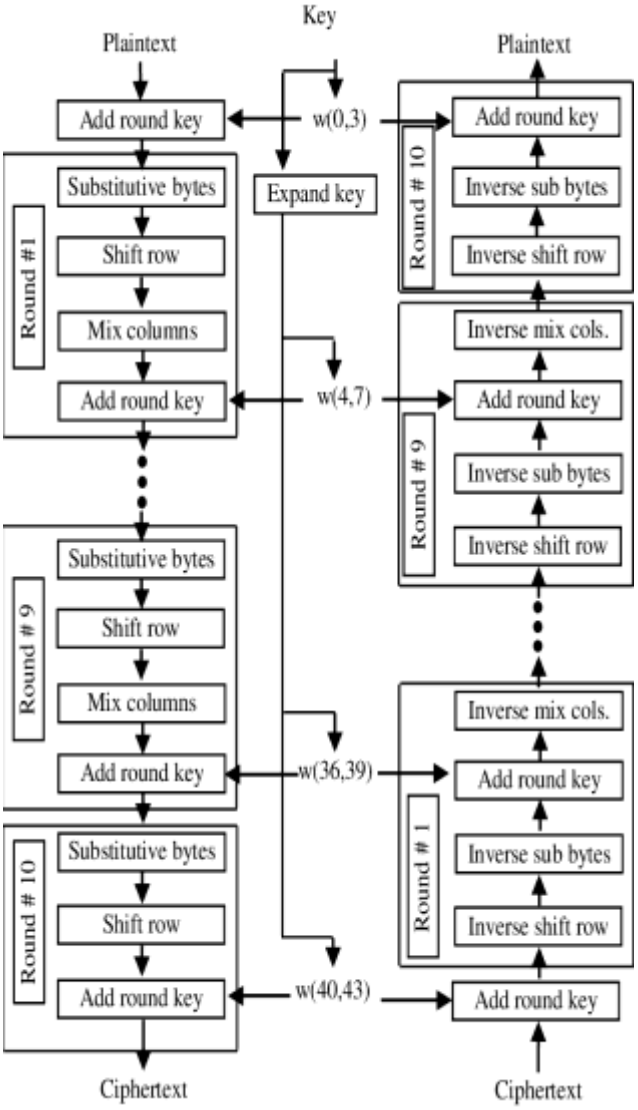
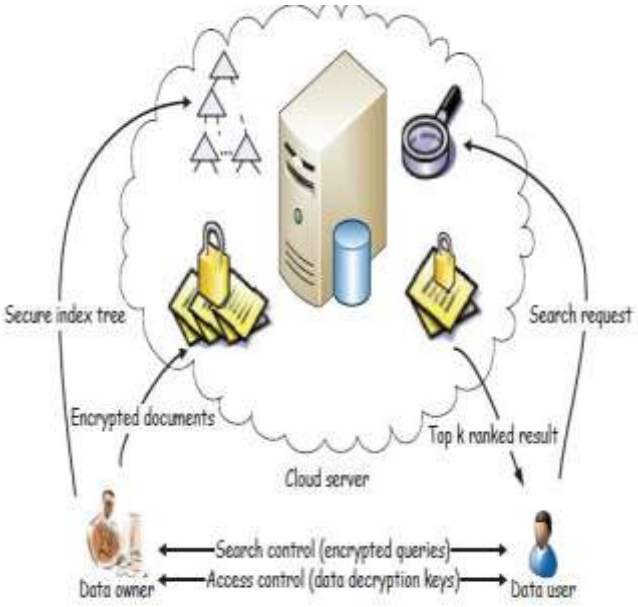


Fig. AES algorithm

3. PROPOSED SYSTEM

3.1 System Development:

Major objective of the proposed system is to extract the top relevant data similar to users query as well as to enhance experience of users while searching. Single keyword searching technique generates unwanted traffic of data which is not that much related to users query. Hence we need to develop robust system which will provide efficient results with multi-keyword searching. As cloud storage is not trusted to keep our plaintext data, we need to save data in encrypted form to ensure privacy. User will register on cloud storage by registration process. Key generation center (KGC) will provide login credentials to user. After successful login user can store their data on cloud. User can upload his confidential data and documents on cloud. Next time whenever user will ask for uploaded document, it will retrieve in encrypted form. So that only user of data is able to decrypt the encrypted data with key. Once document get uploaded on cloud server, proposed system extracts attributes of files such as file name, file date, author name, title of paper, etc,. Results are ranked and retrieved using indexing techniques. Lucene indexing is efficient among other indexing techniques and hence used in the proposed system. Top relevant results are also retrieved matching with user queries using top-k query.



Proposed system will solve the problem of searching of information from encrypted data stored on cloud with the help of multi-keyword search query which also preserves system-wide privacy in cloud computing. Proposed system used Keyword relevance technique as an intermediate similarity measure which extracts keywords from search query to match with keywords stored on cloud. Ranking system is developed which provides the effective retrieval of results using lucene indexing. Top matching results are retrieved using top-k query so that unwanted network traffic gets avoided. Results relevant to user's interest are retrieved efficiently using multiple keyword search procedure. Single Keyword search query will yield too much unnecessary results so that there is necessity of supporting multi-keyword query.

3.2 Lucene Indexing:

When person wants to upload his official or personal data on the cloud server, keywords are get abstracted from that documents and index will be generated for all abstracted keywords [1], [4]. Hashed values of keywords are stored as index. When document will get added and removed from cloud storage, index will get restructured accordingly. In proposed method, for generating index of keywords extracted from documents, Lucene indexing technique is used. Lucene possesses full-text search engine architecture which delivers complete query and indexing engine. Lucene

indexing technique delivers distinct advantages such as indexing is like file format and application platform independent. It also delivers searching of data over documents [9]. A Lucene technique provides continuous updations into indexes of fetched keywords when they are get added and removed from storage space.

Lucene is very wealthy and potent full-text search library written in Java. Lucene indexing is utilized to deliver fulltext indexing over both database objects and documents in various formats. Lucene indexing API is independent from other file formats. Textual data from various resources such as pdf, HTML, Microsoft word, openDocument is get extracted first. In Lucene indexing, to construct an index, whole document gets scan first so as to produce list of postings. Occurrence of word in a document is get described by postings [10]. Posting generally consist of word, document identifier and frequency of the word within that document. Indexing process is one of the main functionality delivered by Lucene. Figure shown below describes the overall indexing procedure carried out by lucene and use of each class. IndexWriter is the most significant and main module of the indexing process.

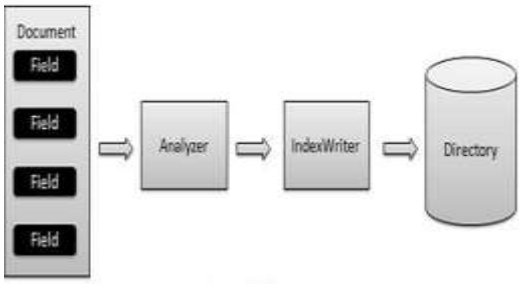
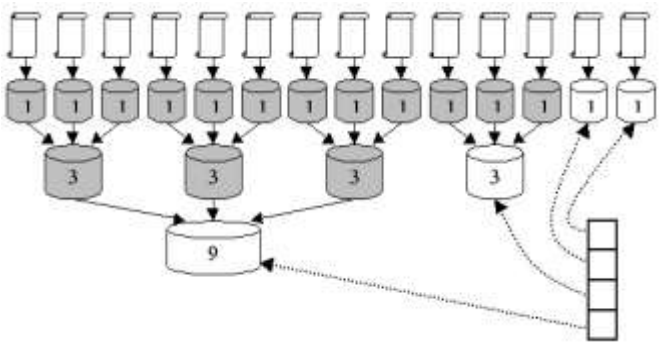


Fig. Lucene Indxing Architecture

Supporting full-text search using Lucene indexing executes two steps:

(1) Generating a lucene index on the documents and on database objects and

(2) Parsing the user query and looking up the prebuilt index to answer the query.



Above figure describes complete working procedure of Lucene. As soon as the document will gets uploaded on to the cloud server index segment will get generated. In accordance with merge factor, index segments will get merged. Index contains hashed values of keywords.

3.3 Top-K Query:

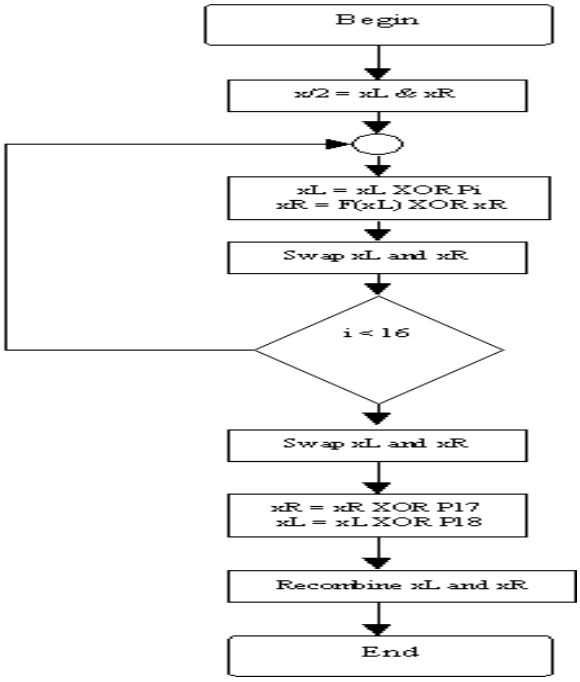
Information retrieval systems utilize diverse approaches to rank query answers. Users are more apprehensive for the most significant i.e., top-k query response from large answer space. Some emerging applications claim for support of top-k queries. In case of the Web, top-k query can be used to provide effectiveness and efficiency of Meta search engines. Distinct applications are also presents in fields of information retrieval and data mining. Goal of this top-k query algorithm is to fetch top matching results from large record set. Top-k query helps to search more precise answers from specified record set that equivalent with filtering keyword, and assemble relevant answers in accordance with their scores. Steiner tree is constructed with each result set which is basically an XML tree [3]. The constructed Steiner tree will holds the list of all records which are already assembled in accordance with their scores. Top-K query algorithm utilizes scores documents against keywords. This algorithm is used to rank "Tuple Units". A tuple unit consists of a set of similar tuples which again consists of query keywords. These tuple units are used to construct tuple set. When two tuples are directly associated with each other then they are merges into one single tuple. Direct scoring and indirect scoring techniques are used in top-k approach to rank every tuple. Top-K query scores every tuple according to two scoring methods, namely, Direct Scoring and Indirect Scoring. Direct Scoring depends on TF-IDF algorithm. TF-IDF stands for "Term Frequency, Inverse Document Frequency".

TF-IDE presents a way to score the words depending on how repeatedly they come into sight across various documents. Scores are termed according to the following criteria: 1) If a word appears frequently in a tuples, it is termed as important and is scored high. 2) If a word appears in many tuples, it is termed as a unique identifier and is scored low. Therefore, common words like "the" and "for", which appear in many tuples, will be neglected. Words that appear frequently in a single tuple will be scale up. The second type of scoring method is Indirect Scoring. Indirect Scoring scores a tuple unit based on the keyword that is indirectly present in the tuple unit. The keyword is scrutinized against each tuple unit for indirect similarity.

3.5 Blowfish Algorithm:

Encryption approaches are categorized in two parts as symmetric key encryptions and public key encryptions. Blowfish falls under Symmetric algorithms. Blowfish algorithm is developed by Bruce Schneider and also known as block cipher. As name in indicated, it divides message into

fixed length blocks while encryption and decryption process. Blowfish algorithm can be used as quick alternative for some existing encryption processes and as it delivers robustness, none of the attack becomes successful against it. Blowfish algorithm consists of key expansion a technique which makes it more complex to break. It also takes less time for encryption process as compared to other encryption algorithms as AES, DES.



Blowfish contains 16 rounds. Each round holds XOR operation and a function. Each round consists of key expansion method and data encryption process [6]. Key expansion generally used for generating initial contents of one array and data encryption uses a 16 round Feistal network. Foundation of Blowfish Algorithm is Feistel Network which iterates simple encryption function 16 times [12]. Proposed system employ Blowfish encryption algorithm for encryption of data. Homomorphic encryption is implemented along with Blowfish algorithm. Homomorphic encryption permits computations on cipher text directly. So that, generated encrypted outcome when again gets decrypted will exactly be similar because all operations are carried out on plaintext. Homomorphic encryption provides facility to perform complex mathematical computations on encrypted data without negotiating the encryption data quality. Blowfish algorithm consists of 64-bit block cipher and variable length key. This algorithm requires less memory space and yields high speed as compared to other algorithms. This algorithm needs 32 bit microprocessor at a rate of one byte for every 26 clock cycles. It employs variable length key block cipher up to 448 bits. Blowfish contains total 16 rounds. Plain text and key are the inputs provided for Blowfish algorithms.

4. CONCLUSION

Proposed system focused on how the user can effectively accumulate their private 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-matching 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.

5. REFERENCES:

[1] Miss Deepali D. Rane, Dr. V. R. Ghorpade , "Multi-User Multi keyword Privacy Preserving Ranked Based Search over Encryted Cloud Data", International Conference on Pervasive Computing (ICPC).

[2] Prachi Jain, Prof.Shubhangi Kharche Effectuation of Blowfish Algorithm using Java Cryptography", International Journal of Scientific & Engineering Research Volume 4, Issue 5, May-2013.

[3] 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

[4] 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

[5] Yanjiang Yang, "Towards Multi-User Private Keyword Search for Cloud Computing" IEEE 4th International Conference on Cloud Computing. 2011.

[6] 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.

[7] Twofish : A 128 Bit Block Cipher by Bruce Schneier, John Kelseyy, Doug Whitingz David, Wagnerx Chris Hall . [8] "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. [10] QIAN Liping, WANG Lidong, "An Evaluation of Lucene for Keywords Search in Large-scale Short Text Storage" IEEE Transaction 2010.

[9] Govind S.Pole, Madhuri Potey, "A Highly Efficient Distributed Indexing system based on large cluster of commodity machines" IEEE Transaction 2012.

[10] Ms Neha Khatri – Valmik, Prof. V. K Kshirsagar, "Blowfish Algorithm", IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278- 0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. X (Mar-Apr. 2014), PP 80-83.