Cloud Based Secured Multi Keyword Ranked Search for Multi-Ownered

Sachin Mane[1], Pragati Mohite[1], Vaibhavi patil [1] and Nidhi Sharma [2]

[1] UG Scholar, [2] Professor

Department of Computer Engineering

Bharati Vidyapeeth college of Engineering

Navi Mumbai, Maharashtra

India

ABSTRACT

For the purpose of economic saving as well as to achieve flexibility users outsourced their data on public cloud. Before storing data on cloud user wish to apply some security constraints to protect their data form malicious attacks. Therefore, they used to refer encryption technique to encode their data and then they will outsource it. Existing systems which refers encryption technique are inefficient as they are having only single keyword search over large document. To the best solution for this inefficiency we proposed multikeyword search strategy for large number of documents. Our multikeyword search technique is efficiently word against the searching of multiple keywords simultaneously. It may be referred as lightweight process of searching. In our system, we used two protocols to generate a secret key dynamically as well as for vali user authentication. Our system effectively works for large number of databases and multiple keyword searching. In our system, we provide higher privacy requirements to implement the data security for cloud stored data. Our main contribution in this project is to give backup and restore facility at the user end and mainly we focused on multikeyword searching with ranking approach.We give an overview of our framework for keyword searching with summaries, besides we describe a ranking algorithm for ranked keyword search and their results. Keyword searches are typically done so that users can actively utilize clouds to query a collection.

KeyWords: ***M****ultiple ranked keyword search, Multiple owners, Privacy preserving system, Dynamic secret key.*

1. INTRODUCTION

Cloud computing provides multiple facility such as, to protect sensitive data, like emails, governments files, employee personal records in various sectors etc. In cloud computing, privacy is provided for accessing data, to outsourced the data and also to maintain flexibility of data [1]. By using the concept of virtualization and firewall CSP gives guarantee of data privacy to the data owner in cloud system. CSP has full control on cloud hardware, software as well as on data owner’s [3]. Previously used technique that is data encryption suffered from many challenges such as inefficiency, costing when there is large amount of data is present [1]. Due to single keyword search strategy, it is most time-consuming task and hence it results into inefficiency [7]. J. Li, Q. Wang introduced fuzzy multi-keyword search technique over an encrypted data [9]. Whereas Boolean keyword search provides the solution for ranked keyword search for large dataset [11].

We proposed multikeyword search technique for ranked keywords where large number of encrypted data is available. From security perspective, our system provides unique ID for users. This ID is unique or hidden from CSP and third party user. Therefore, better privacy and data confidentiality is achieved with our proposed system. Our system works effectively where there is need to protect data sensitivity. Some With implementation of such type of system we aim to provide a backup and restore facility at users end and ranked multikeyword search over large database in cloud. Our system gives the best solution against single keyword search over large amount of cloud data. And also, it proves its efficiency as well as security.

2. PROPOSED SYSTEM

Above we represent system architecture

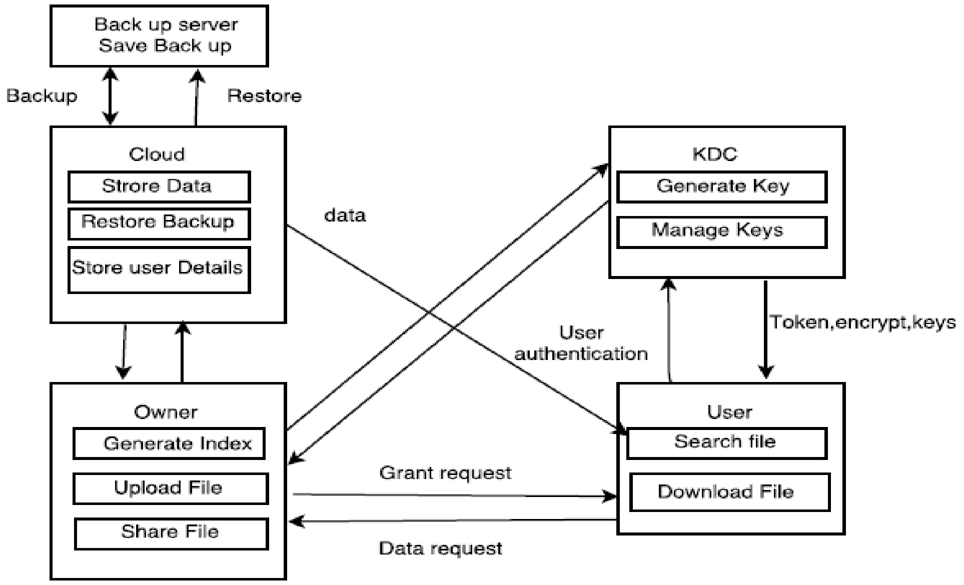

Fig 1.1 System Architecture.

1. Register:

Step 1: User Registration

Step 2: During registration user provides some necessary information.

Step 3: Token is given to end user by administrative server provide.

2. Upload File:

Step 1: Login. Step 2: Upload file.

Step 3: Administrative Server provides an encryption key to the user.

Step 4: Index of files for user.

Step 5: Encrypt user index.

Step 6: To encrypt index SHA-1 algorithm is used.

Step 7: Encrypt document AES algorithm

3. Share document with another User:

Step 1: Access privileges to data structure present on Administrative Server.

4. Search:

Step 1: User login that verified by Administrative Server.

Step 2: Get key from Administrative Server.

Step 3: Generate Trapdoor for search. Step 4: Achieved result set.

5. Backup:

Step 1: User login that verified by Administrative Server.

Step 2: User get key from Administrative Server.

Step 3: Show own files.

Step 4: Select file and use backup facility where last modified copy is saved.

6. Restore:

Step 1: User login and verified by Administrative Server.

Step 2: User get key from Administrative Server.

Step 3: Show own deleted files. Step 4: Select file for restore.

3.ALGORITHM

AES (Advanced Encryption Standard)

AES is based on a design principle known as a substitution-permutation network, a combination of both substitution and permutation, and is fast in both software and hardware. Unlike its predecessor DES, AES does not use a Feistel network. AES is a variant of Rijndael's which has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, the Rijndael's specification *per se* is specified with block and key sizes that may be any multiple of 32 bits, both with a minimum of 128 and a maximum of 256 bits.

AES operates on a 4 × 4 column-major order matrix of bytes, termed the *state*, although some versions of Rijndael's have a larger block size and have additional columns in the state. Most AES calculations are done in a particular finite field.

The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the ciphertext. The number of cycles of repetition are as follows:

- 10 cycles of repetition for 128-bit keys.
- 12 cycles of repetition for 192-bit keys.
- 14 cycles of repetition for 256-bit keys.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

High-level description of the algorithm

1. KeyExpansions—round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate 128-bit round key block for each round plus one more.

2.InitialRound

a. AddRoundKey—each byte of the state is combined with a block of the round key using bitwise xor.

3.Rounds

a. Sub Bytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.

b. ShiftRows—a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.

C .MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.

d. AddRoundKey.

4. Final Round (no MixColumns)

a. SubBytes

b. Shift Row

c. AddRoundKey

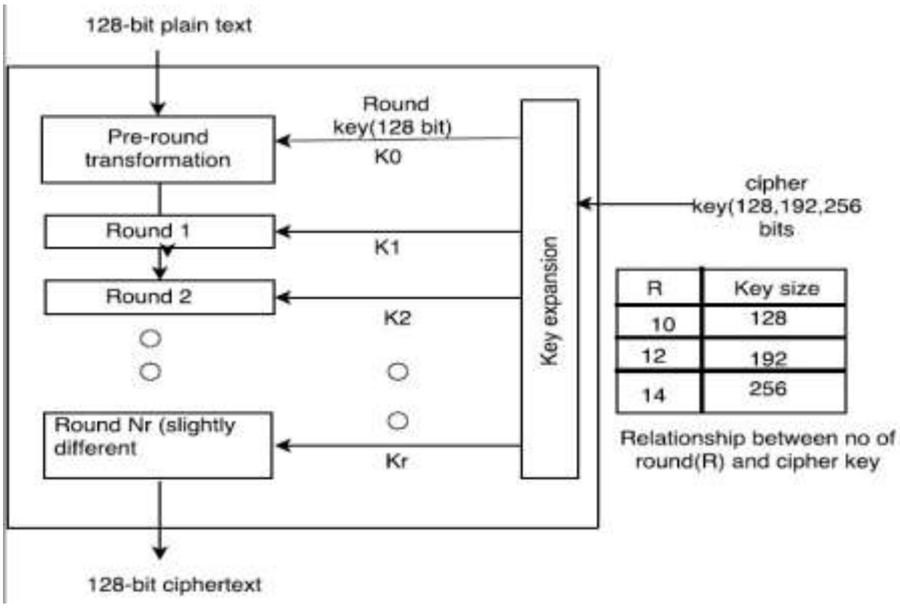

Fig 1.2 AES structure

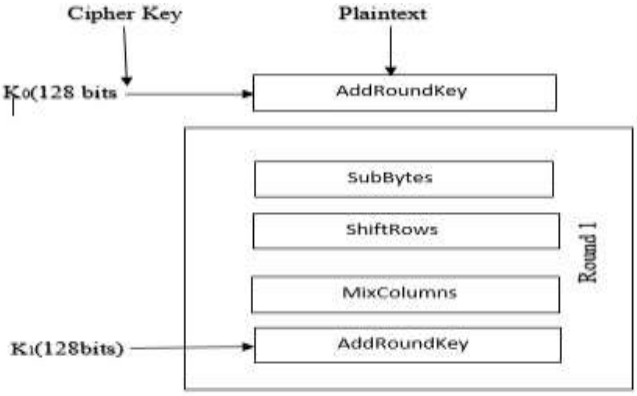

Fig 1.3 AES Decryption

Ranking Algorithm:

[1] Define the list of available clouds on any public cloud server called Cloud (1), Cloud (2) ……..Cloud(n)

[2] For i=1 to n {Identify parameters for Cloud (i) called Availability (i), Response Time(i), Security(i)}

[3] Accept the User Query called Req under the specification ReqKeyword, ReqSecurity, ReqDeadLine,

[4] Activate the Middle layer to provide the best service selection

[5] Accept the user query and filter it to retrieve the keywords under the following step

- Remove the stop list words from the query list
- Rank the different keywords respective to category
- Find the frequency of keywords
- Keep the most occurring keywords and present as relevancy measure

[6] As the keywords retrieve perform query on each public cloud and perform the content and tag based match.

[7] Find the list of M clouds that satisfy the relevancy criteria as well as identify the other cloud parameters like response time, security measure

[8] For i=1 to M [Perform the Content based similarity measure as]

[9] RelevencyVector = 0 For j=1 to Length (User Keywords) {RelevencyVector=RelevencyVector + Keyword Occurrence(Cloud(i), Keyword(j)) /Total Keywords(Cloud(i), Keyword (j));}

[10] Security Vector=0; If (UserSecurityReq=Security (Cloud(i)) Security Vector=1;

[11] ResponseTimeVector=0 If (User Deadline>Response Time(Cloud(i))
Input: User Query Output: Keyword Extraction

Input: Keyword Extraction Query Output: Ranked cloud list Ranked Keyword Search in Cloud{ResponseTimeVector=UserDeadline-ResponseTime(Cloud(i)}

[12]Rank(Clound(i))=RelevencyVector*w1+SecurityVector*w2 +ResponseTimeVector*w3;

[13] As user get Ranked list of clouds, selection can be performed for best cloud service provider respective to user interest.

4.CONCLUSION

Our multi-keyword ranked search system provides encrypted cloud data that utilizes efficient similarity measure of coordinate matching. Previous systems mainly focused on providing privacy to the data on cloud. We provide more real privacy system. In our system, stringent privacy is provided by allocating a unique ID to cloud user. Our system hides this user ID from the cloud service provider as well as the third-party user, to protect the user’s data on cloud from the CSP and the third-party user. Dropbox is the cloud use for storing and retrieving the data of our system.

To maintain data confidentiality, to hide user’s identity may work efficiently. Also, multiple owners for data are concept newly introduced in our system. While uploading the data index terms are generated that helped for indexed search which is efficient one. In contribution, we provide an efficient backup and restore facility to the end user.

REFERENCES

[1] Wei Zhang, Student Member, IEEE, Yaping Lin, and Siwang Zhou," Privacy Preserving Ranked Multi-Keyword Search for Multiple Data Owners in Cloud Computing "Hong Kong in 2005

[2]. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Apr 2010.

[3]. C. Wang, S. Chow, Q. Wang, K. Ren, and W. Lou, "Privacypreserving public auditing for secure cloud storage," Computers, IEEE Transactions on, vol. 62, no. 2, pp. 362–375, 2013.

[4]. D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proc. IEEE International Symposium on Security and Privacy (S&P'00), Nagoya, Japan, Jan.2000, pp. 44–55.

[5]. P. Golle, J. Staddon, and B. Waters, "Secure conjunctive keyword search over encrypted data," in Proc. Applied Cryptography and Network Security (ACNS'04), Yellow Mountain, China, Jun. 2004, pp. 31–45.

[6]. C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure ranked keyword search over encrypted cloud data," in Proc. IEEE Distributed Computing Systems (ICDCS'10), Genoa, Italy, Jun. 2010, pp. 253–262.

[7]. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacypreserving multi-keyword ranked search over encrypted cloud data," in Proc. IEEE INFOCOM'11, Shanghai, China, Apr. 2011, pp. 829–837

[8]. J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy keyword search over encrypted data in cloud computing," in Proc. IEEE INFOCOM'10, San Diego, CA, Mar. 2010, pp. 1–5.

[9]. P. Xu, H. Jin, Q. Wu, and W. Wang, "Public-key encryption with fuzzy keyword search: A provably secure scheme under keyword guessing attack," Computers, IEEE Transactions on, vol. 62, no. 11, pp.2266–2277, 2013.

[10]. B. Wang, S. Yu, W. Lou, and Y. T. Hou, "Privacypreserving multi-keyword fuzzy search over encrypted data in the cloud," in IEEE INFOCOM, Toronto, Canada, May 2014, pp. 2112–2120.

[11]. C. Wang, N. Cao, K. Ren, and W. Lou, "Enabling secure and efficient ranked keyword search over outsourced cloud data," Parallel and Distributed Systems, IEEE Transactions on, vol. 23, no. 8, pp. 1467–May 2013.

ACNOWLEDGEMENT

We would like to express our heartfelt thanks to all those who have helped us in making this report a success. However, it would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them.

We are highly indebted to DR.D.R. Ingle for his guidance and also thanks to Prof. Nidhi Sharma for her guidance and constant supervision as well as to all the staff members of Computer Dept. and lab facilities by which we have enlighten on topics regarding our report.

We would like to express our heartfelt thanks to all those who have helped us in making this report a success. However, it would not have been possible without the kind support and help of many individual literature. We would like to extend our sincere thanks to all of them

Our thanks and appreciations also go to our colleagues in developing the report and people who have willingly helped us out with their abilities.