International Journal of Scientific Research in Computer Science, Engineering and Information Technology

© 2017 IJSRCSEIT | Volume 2 | Issue 6 | ISSN : 2456-3307

An Efficient Storage and Retrieval Method Based On Multi-Key Ranked Search & Improved Hierarchical Clustering Index for Cloud Data

Ajeet Mishra[1], Prof. Umesh Kumar Lilhore[2], Prof. Nitesh Gupta[3]

M. Tech. Research Scholar[1], Head PG[2], Assistant Professor[3]

NRI Institute of Information Science & Technology Bhopal, Madhya Pradesh, India

ABSTRACT

In this current scenario, computer technologies are getting change day by day. The cost of computing resources are extremely high and it is quite difficult to upgrade hardware's software. Now users are demanding more innovative technologies which can provide optimum utilization of computing resources and cloud computing is one of them. Cloud computing in an improved form of various existing technologies such as grid computing, cluster computing, and distributed computing. Cloud computing serves computing resources such as PaaS, IaaS, and SaaS to cloud users on demand and 'pay peruses' based. A cloud user can store their private and essential data over the cloud and can retrieve any time. Day by day the number of cloud users and the size of cloud data are getting increases. Cloud service providers have to ensure the data privacy and integrity as well availability of stored data. Various cloud researchers are working on cloud data security and efficient retrieval. In this work, we are presenting an efficient data storage and retrieval method for encrypted data based on multi-keyword ranked search by improved hierarchical clustering index for cloud data to improve cloud performance. The proposed method basically takes place in two phases. In first phase IAES-256 bit data encryption and decryption methods are used to maintain data security and SHA-1 method is used to calculate the hash values of the message to maintain the data integrity and second phase uses efficient data retrieval method based on multi-keyword ranked search by improved hierarchical clustering index (by dynamic K-mean) with bagging approach for cloud data. Bagging provides a predictive probabilistic model which reduces the noise and irrelevant data during classification, which improves the accuracy. The proposed method uses an "In order" to verify the authenticity of search results, a structure called minimum hash sub-tree. Proposed method (MRSE-IHCI With bootstrap) and existing method (MRSE-HCI) both are implemented and compared based on various performance measuring parameters such as encryption time and storage, retrieval time and search time. Experimental result analysis clearly shows that proposed method performs outstanding over existing data storage and retrieval method for cloud data.

Keywords : Cloud Computing, Data Security, Data Retrieval, IAES, MRSE, IHCI, Bagging.

I. INTRODUCTION

Cloud computing is the since a long time ago imagined a vision of registering as a utility, where cloud clients can remotely store their information into the cloud in order to appreciate the excellent systems, servers, applications, and administrations from a common pool of configurable processing assets [2]. The upsides of cloud computing incorporate on-request self-benefit, universal system get to, an area free asset pooling, fast asset flexibility, use-based estimating, transference of hazard, and so forth. It's awesome adaptability and monetary reserve funds are persuading the two people and ventures to outsource their nearby complex information administration framework into the cloud. Information ruptures of critical cloud benefits additionally show up every once in a while. Additionally, the cloud specialist co-ops may likewise intentionally look at clients' information for different inspirations. Hence, we contend that the cloud is naturally neither secure nor dependable from the view purpose of the cloud clients [1, 4]. Without giving solid security, protection, and unwavering quality certification, it is difficult to anticipate that cloud clients will turn over control of their information to cloud servers exclusively in view of financial funds and administration adaptability. Ranked search can also elegantly eliminate unnecessary network traffic by sending back only the most relevant data, which is highly desirable in the “pay-as-you-use” cloud paradigm. For privacy protection, such ranking operation, however, should not leak any keyword related information [21]. Prior cloud storage systems based on erasure codes or replication techniques have either high decoding computational cost for data users or too much burden of data storage and repair cost for data owners. This entire work mainly focused on cloud data security and integrity. In this work, we are presenting an efficient data storage and retrieval method for encrypted data based on multi-keyword ranked search by improved hierarchical clustering index for cloud data to improve cloud performance [15].

II. CLOUD COMPUTING

Cloud computing is a method or technique for enabling convenient, on-demand network access to a shared pool of computing resources (such as computer networks, servers, applications, storage, and services) [10,17] that can be continuously provisioned and released with minimum management efforts. The main underlying idea behind the cloud computing technology is the separation of applications from the layer of operating systems and the hardware on which they will run [18].

2.1 Types of Cloud-Cloud deployment models are [7]

Public cloud- A public cloud is one standard of cloud computing, in which a cloud service provider provides a virtual environment, make a pool shared resources, such as applications and storage, offered to the general public over the web.

Private cloud- Organizations choose to build their private cloud as to keep the strategic, operation and other reasons to themselves and they feel more secure to do it.

Hybrid model- It consists of multiple service providers. It provides the services of both public and private cloud. It is used by organizations when they need both private and public clouds both.

2.2 Cloud Computing Services- Users can access these services in a pay per use on-demand model. Cloud computing provides following services.

IaaS- In Infrastructure as a service, the service provider shares infrastructure resources to support operations done by the end user.

PaaS-This service is used to complete the life cycle of building and delivering web applications, which are available over the internet.

SaaS- The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure.

III. EXISTING WORK IN CLOUD DATA SECURITY & RETRIEVAL

Existing systems are based on following data retrieval schemes [12]-

3.1 Boolean Keyword Search-Boolean systems allowed customers to specify their information need using a combination of Boolean operators AND, OR and NOT.

3.2 Single Keyword Based Searchable Encryption-In single keyword searchable encryption schemes encrypted searchable indexes are uses and its contents hidden to the server unless it’s given appropriate trapdoors generated via secret key(s)[3].

3.3 Searchable Encryption- It allows users to securely search complete encrypted data through keywords. This method supports only Boolean search, without capturing any relevant data [2].

3.4 MKSE or Multi Keyword Searchable Encryption- In cloud computing to search functionalities, conjunctive keyword search over encrypted data had been proposed. These schemes incur large overhead caused by their fundamental primitives, such as computation cost by bilinear map [5].

3.5 TOP-K data retrieval method- In order to efficiently solve data search problem, existing self-indexing algorithms are not sufficient [14]. Two approaches to enhance the retrieval of self-index have been proposed. The first uses a document-based array, which uses to map in between every suffix in set T to its corresponding document identifier.

3.6 Ranked Keyword Search- Ranked search greatly enhances system usability by returning the matching files in a ranked order regarding certain relevance criteria (eg. keyword frequency) thus; making one step closer toward practical deployment of privacy-preserving data hosting services in the context of cloud computing [6].

3.7 Fuzzy Keyword Searchable Encryption: Fuzzy keyword search greatly enhances system usability by returning the matching files when users searching inputs exactly match the predefined keywords or the closest possible matching files based on keyword similarity semantics when an exact match fails [5].

3.8 Plaintext Fuzzy Keyword Search- The importance of fuzzy search has received attention in the context of plaintext searching in information retrieval community [12].

IV. PROBLEM STATEMENT & OBJECTIVE

Cryptographic methodologies can accomplish the security objectives for the cloud framework, it may altogether decrease the effectiveness of the cloud framework and thus influences sending of customary information usage to benefit troublesome. Various cloud researchers are working on cloud data security and efficient retrieval of these encrypted data.

4.1 This research mainly deals with two major issues in cloud computing-

Data storage – In cloud computing secure and efficient data storage are always challenging. Encryption methods are used for secure data storage, Which encounters with several issues such as-

- Encryption & Decryption Time
- Auditing time and Storage

Data Retrieval – In Cloud storage user's data is stored in encrypted form. This encrypted data are travels over the networks from one client to another. Stored encrypted data encounters with several issues such as-

- Retrieval time
- Data Searching Time
- Transmission Time

Objective of the work- In this work, we are presenting an efficient data storage and retrieval method for encrypted data based on multi-keyword ranked search by improved hierarchical clustering index for cloud data to improve cloud performance**.**

This research work mainly deals with following issues-

- Higher encryption and decryption time
- Higher searching time
- Higher retrieval time
- Higher ranking and indexing time

V. PROPOSED SOLUTION

In this work, we are presenting an efficient data storage and retrieval method for encrypted data based on multi-keyword ranked search by improved hierarchical clustering index for cloud data to improve cloud performance. The proposed method basically takes place in two phases. In first phase IAES-256 bit data encryption and decryption methods are used to maintain data security and SHA-1 method is used to calculate the hash values of the message to maintain the data integrity and second phase uses efficient data retrieval method based on multi-keyword ranked search by improved hierarchical clustering index (by dynamic K-mean) with bagging approach for cloud data. Bagging provides a predictive probabilistic model which reduces the noise and irrelevant data during classification, which improves the accuracy. The proposed method uses an "In order" to verify the authenticity of search results, a structure called minimum hash sub-tree.

5.1 WORKING OF THE PROPOSED SYSTEM

- In this work, we are presenting an efficient data storage and retrieval method for encrypted data based on multi-keyword ranked search by improved hierarchical clustering index for cloud data to improve cloud performance. The proposed method basically takes place in two phases.
- In first phase IAES-256 bit data encryption and decryption methods are used to maintain data security and SHA-1 method is used to calculate the hash values of the message to maintain the data integrity and second phase uses efficient data retrieval method based on multi-keyword ranked search by improved hierarchical clustering index (by dynamic K-mean) with bagging approach for cloud data.
- Bagging provides a predictive probabilistic model which reduces the noise and irrelevant data during classification, which improves the accuracy.

5.2 Architecture of proposed MRSE-IHCI- Proposed Method has following components.

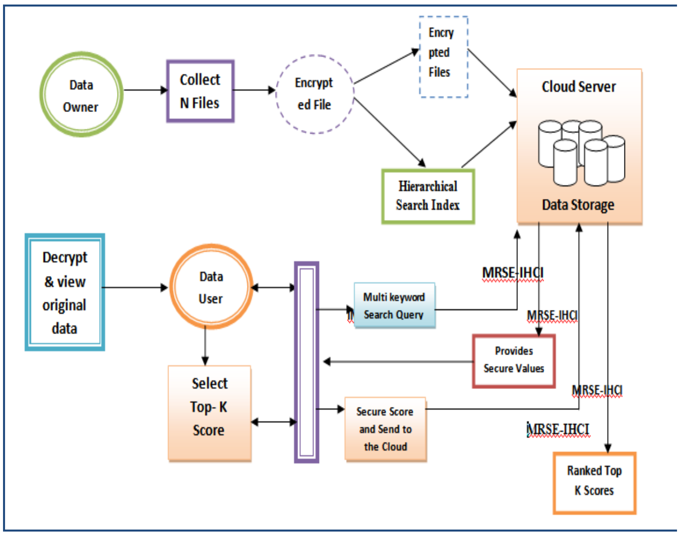


Figure 1. Architecture of proposed MRSE-IHCI

- **Index (D; sk)→ I:** Encrypted index is generated in this phase by using the above mentioned secret key. At the same time, the clustering process is also included current phase.
- **Keygen ($1^{l(n)}$)→(sk; k):**It is used to generate the secret key to encrypt index and documents.
- **Enc (D; k) → E:** The document collection is encrypted by a symmetric encryption algorithm which achieves semantic security.
- **Trapdoor (w; sk)→Tw:** It generates encrypted query vector Tw with users input keywords and secret key.
- **Search (Tw; I; ktop)→ (Iw; Ew):** In this phase, cloud server compares trapdoor with an index to get the top-k retrieval results.
- **Dec (Ew; k)→ Fw:** The returned encrypted documents are decrypted by the key generated in the first step.

5.3 PROPOSED ALGORITHM

Phase-1 (Data Secure)

5.3.1 ***Key Generation Module****- Steps in the algorithm:*

Step-1 *Sender and Receiver agree on a prime number p and a base g.*

Step-2 *Sender chooses a secret number a, and sends Receiver (g^a mod p).*

Step-3 *Receiver chooses a secret number b and sends Sender (g^b mod p).*

Step-4 *Sender computes ((g^b mod p)^a mod p)*

Step-5 *Receiver computes ((g^a mod p) ^{b} mod p), Both Sender and Receiver can use this number as their key. Notice that p and g need not be protected.*

5.3.2 IAES-256 Encryption Module-

Step-1 Key Expansions*-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.*

Step-2 initial round

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

Step-3 Rounds

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

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

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

3. 4 AddRoundKey

Step-4 Final Round (no MixColumns)

4.1 SubBytes

4.2 ShiftRows

4.3 AddRoundKey.

5.3.3 SHA-1 (Hash Generation Module)

Step 1: Append Padding Bits*-The message is "padded" with a 1 and as many 0's as necessary to bring the message length to 64 bits less than an even multiple of 512.*

Step 2: Append Length*-64 bits are appended to the end of the padded message. These bits hold the binary format of 64 bits indicating the length of the original message.*

Step 3: Prepare Processing Functions*- SHA1 requires 80 processing functions defined as:*

f(t;B,C,D) = (B AND C) OR ((NOT B) AND D) (0 <= t <= 19)

f(t;B,C,D) = B XOR C XOR D (20 <= t <= 39) f(t;B,C,D) = (B AND C) OR (B AND D) OR (C AND D) (40 <= t <=59)

f(t;B,C,D) = B XOR C XOR D (60 <= t <= 79)

Step 4: Prepare Processing Constants*- SHA1 requires 80 processing constant words defined as:*

K(t) = 0x5A827999 (0 <= t <= 19)

K(t) = 0x6ED9EBA1 (20 <= t <= 39)

K(t) = 0x8F1BBCDC (40 <= t <= 59)

K(t) = 0xCA62C1D6 (60 <= t <= 79)

***Step 5: Initialize Buffers**- SHA1 requires 160 bits or 5 buffers of words (32 bits): H0 = 0x67452301, H1 = 0xEFCDAB89,* H2 = *0x98BADCFE , H3 = 0x10325476 , H4 = 0xC3D2E1F0*

***Step 6: Processing Message** in 512-bit blocks (L blocks in total message)- This is the main task of a SHA1 algorithm which loops through the padded and appended message in 512-bit blocks.*

Input and predefined functions:

***M[1, 2, ..., L]:** Blocks of the padded and appended message f(0;B,C,D), f(1,B,C,D), ..., f(79,B,C,D): 80 Processing Functions K(0), K(1), ..., K(79): 80 Processing Constant Words*

H0, H1, H2, H3, H4, H5: 5 Word buffers with initial values

Phase-2 (Searching and Retrieval)

5.4 (MRSE-IHCI)

***5.4.1 Setup** –*

5.4.1.1 The data owner randomly generates a (n + 2)-bit vector as S and

5.4.1.2 Two (n+2)×(n+2) invertible matrices {M1,M2}.

5.4.1.3 The secret key SK is in the form of a 3-tuple as {S,M1,M2}.

5.4.2 BuildIndex (F, SK)-

5.4.2.1 The data owner generates a binary data vector Di, for every document Fi, where each binary bit Di[j] represents whether the corresponding keyword Wj appears in the document Fi.

5.4.2.2 Dimension extending and splitting procedures on Di.

5.4.2.3 These procedures are similar with those in the secure kNN computation except that the (n + 1),th entry in $\vec{D}$ i is set to a random number ε i, and the (n + 2)-th entry in $\vec{D}$ i is set to 1 during the dimension extending.

5.4.2.4 $\vec{D}$ I is therefore equal to (Di, εi, 1).

5.4.2.5 Finally, the sub-index, Ii = {$M^{T}{}_{1}$ →Di', $M^{T}{}_{2}$ → Di''} is built for every encrypted document Ci.

***5.4.3 Trapdoor (fW)** - With t keywords of interest in fW as input,*

5.4.3.1 One binary vector Q is generated where each bit Q[j], indicates whether Wj ∈ fW is true or false.

5.4.3.2 Q is first extended to, n + 1-dimension which is set to 1, and then scaled by a random number r != 0, and

5.4.3.4 Finally extended to a (n + 2)-dimension vector as $\vec{Q}$ where the last Dimension is set to another random number t.

5.4.3.5 $\vec{Q}$ is therefore equal to (rQ, r, t). After applying the same splitting and

5.4.3.6 Encrypting processes as above, the trapdoor TfW is generated as {M −1 1 $\vec{Q}$ ', M −1 2 $\vec{Q}$ ''}

***5.4.4 Query (TfW, k, I)** with the trapdoor TfW, the cloud server computes the similarity scores of each document Fi as in equation 1.*

5.4.4.1 WLOG, we assume r > 0. After sorting all scores, the cloud server returns the top-k ranked id list FfW. →Note that in the original case, the final score is simply rDi

5.4.5 (Algorithm to create clusters by improved hierarchal clustering method (Dynamic Key Means with bagging))

5.4.5.1 Input the initial set of k clusters with center C

5.4.5.2 Set the threshold Th_{min}

5.4.5.3 While k!= stable

5.4.5.4 Generates a new set of clusters centers C_0

5.4.5.5 Applying bagging to check cluster accuracy by predictive classification for new clusters

5.4.5.6 for every cluster get the minimum relevance score (min_{score})

5.4.5.7 if the $min_{score} < Th_{min}$

5.4.5.8 add a new cluster

5.4.5.9 Repeat (5.4.5.4 to 5.4.5.8) till k is not stable

5.4.6 (Algorithm to Built Minimum hash sub tree)

5.4.6.1 Built minimum Hash sub-tree based on improved hierarchal clustering method

5.4.6.2 For each leaf node i

5.4.6.3 Calculate its hash value

5.4.6.4 While not tree Root

5.4.6.5 For each nonleaf node j

5.4.6.6 Calculate its hash value

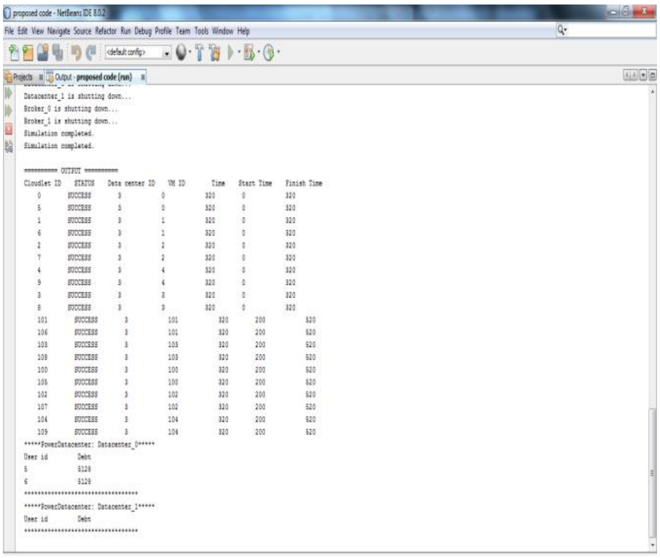
5.4.6.7 Construct node (i d j)

5.4.6.8 Go to the upper level

5.4.6.9 Calculate hash value of Root

VI. SIMULATION RESULTS

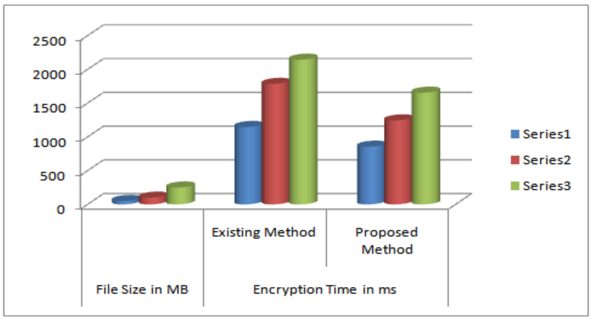
Existing (MRSE-HCI) and Proposed (MRSE-IHCI) both are implemented over cloud sim simulator 3.0. Programs are written in JAVA language.


proposed code - NetBeans IDE 8.0.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Projects
Output - proposed code (run)

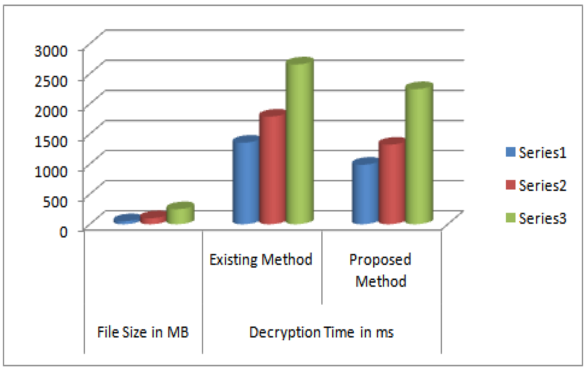
```
Datacenter_1 is shutting down...
Broker_0 is shutting down...
Broker_1 is shutting down...
Simulation completed.
Simulation completed.

========== OUTPUT ==========
Cloudlet ID    STATUS    Data center ID    VM ID      Time    Start Time    Finish Time
    0         SUCCESS        3               0         320      0             320
    5         SUCCESS        3               0         320      0             320
    1         SUCCESS        3               1         320      0             320
    6         SUCCESS        3               1         320      0             320
    2         SUCCESS        3               2         320      0             320
    7         SUCCESS        3               2         320      0             320
    4         SUCCESS        3               4         320      0             320
    9         SUCCESS        3               4         320      0             320
    3         SUCCESS        3               3         320      0             320
    8         SUCCESS        3               3         320      0             320
    101         SUCCESS        3               101         320        200            520
    106         SUCCESS        3               101         320        200            520
    103         SUCCESS        3               103         320        200            520
    108         SUCCESS        3               103         320        200            520
    100         SUCCESS        3               100         320        200            520
    105         SUCCESS        3               100         320        200            520
    102         SUCCESS        3               102         320        200            520
    107         SUCCESS        3               102         320        200            520
    104         SUCCESS        3               104         320        200            520
    109         SUCCESS        3               104         320        200            520
*****PowerDatacenter: Datacenter_0*****
User id         Debt
5               5128
6               5128
**********************************
*****PowerDatacenter: Datacenter_1*****
User id         Debt
**********************************
```

Figure 2 Simulation result

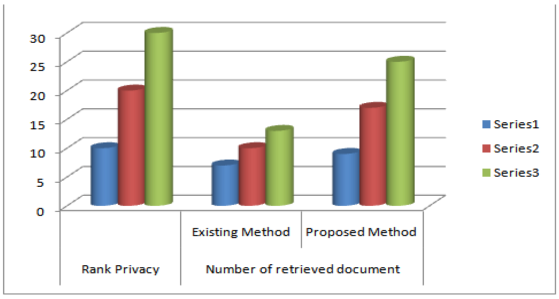
6.1. Data encryption time- This is the total time required to encrypt data and store on a cloud server.



6.2 Data decryption time- This is the total time required to decrypt data and success from the cloud server.



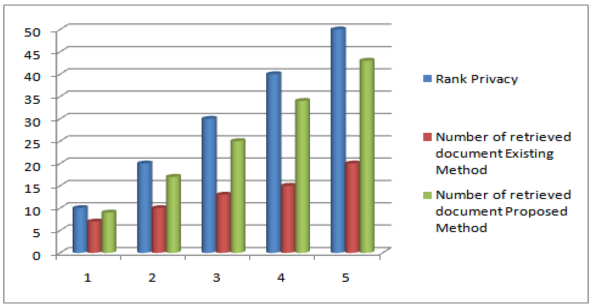
6.3 Data searching time-This is the total time requiring searching query data over cloud server.



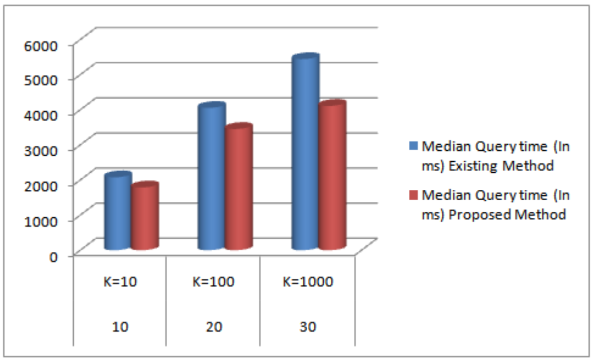
6.4 **Rank Privacy-**Rank privacy can quantify the information leakage of the search results.

$$P_k = \sum_{i=1}^{k} P_i / k$$

Here k= number of Top-K retrieved, Pi= | ci' -ci| , ci'is the ranking of document



6.5 Query time-The time taken to submit a query with difference workload sizes.



Influences- The above graphs 6.1 to 6.5 shows result in comparison of existing (MRSE-HCI) and Proposed (MRSE-IHCI). Above graphs clearly, show that proposed method shows better results over existing method.

VII. CONCLUSIONS & FUTURE WORK

This work proposed an efficient data storage and retrieval method for encrypted data based on a multi-key ranked search by improved hierarchical clustering index for cloud data. We also proposed the MRSE-IHCI architecture to adapt to the requirements of data explosion, online information retrieval, and semantic search. The experiment result proves that the proposed architecture not only properly solves the multi-keyword ranked search problem, but also brings an improvement in search efficiency, rank security, and the relevance between retrieved documents.

In future, we can implement these schemes in real time environment instead of the simulator. More security constants can be added. The proposed scheme can be implemented on real-time dynamic data of more size.

VIII. REFERENCES

[1]. Chi Chen, Xiaojie Zhu, Peisong Shen, J.Hu, S.Guo, Z.Tari, and Albert Y. Zomaya, Fellow, “An Efficient Privacy-Preserving Ranked Keyword Search Method", IEEE Transactions on Parallel and Distributed Systems, the year 2015 PP 1-15.

[2]. V.Sahuarita, S.J.Saritha “A Privacy and dynamic Multi-keyword Ranked Search Scheme over Cloud Data Encrypted", IEEE Year 2016, PP 131-135

[3]. Zhihua Xia, Member, Xingming Sun, and Qian Wang," A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data", IEEE transactions on parallel and distributed systems, vol. 27, no. 2, February 2016, PP 340-353.

[4]. Aashi Qul Huq A, Bhaggiaraj S,"Improving Privacy Multi-Keyword Top-K Retrieval Search Over Encrypted Cloud Data", In International Journal Of Engineering And Computer Science ISSN: 2319-7242, Volume 4 Issue 4 April 2015, Page No. 11385-11390.

[5]. Cong Wang, Qian Wang, Kui Ren, Member, Ning Cao, and Wenjing Lou," Towards Secure and Dependable Storage Services in Cloud Computing", In Proc. the 17th IEEE International Workshop on Quality of Service (IWQoS’09) IEEE 2009, Page No. 999-1013.

[6]. Cheng-Kang Chu, Sherman S. M. Chow, Wen-Guey Tzeng, Jianying Zhou, and Robert H. Deng," Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage", In IEEE Transactions on Parallel and Distributed Systems. Volume: 25, Issue: 2. Year: 2014, Page No. 521-533.

[7]. D. Pratiba, Dr. G.Shobha and Vijaya Lakshmi.P.S," Efficient data retrieval from cloud storage using data mining technique", International Journal on Cybernetics & Informatics (IJCI) Vol. 4, No. 2, April 2015, Page No.271-280.

[8]. Ahmed Shawish and Maria Salama," Cloud Computing: Paradigms and Technologies", in Proc. Inter cooperative Collective Intelligence: Techniques and Applications, Studies in Computational Intelligence 495, Springer-Verlag Berlin Heidelberg 2014, Page No. 642-671.

[9]. Danan Thilakanathan, Shiping Chen, Surya Nepal and Rafael A. Calvo," Secure Data Sharing in the Cloud", In Proc. Security, Privacy, and Trust in Cloud Systems, DOI: 10.1007/978-3-642-38586-5_2, Springer-Verlag Berlin Heidelberg 2014, Page No. 888-921.

[10]. Keiko Hashizume, David G Rosado and Eduardo Fernández," an analysis of security issues for cloud computing", in Proc. Journal of Internet Services and Applications, Springer 2013, Page No.81-84.

[11]. Nelson Gonzalez, Charles Miers, Fernando Red´ıgolo, Marcos Simplıcio, Tereza Carvalho, Mats N Naslund and Makan Pourzandi," A quantitative analysis of current security concerns and solutions for cloud computing", In Proc. Journal of Cloud Computing: Advances, Systems, and Applications, Springer 2012, Page No. 178-196.

[12]. Cong Wang, Ning Cao, Kui Ren, Wenjing Lou," Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data", In Proc. 30th International Conference on Distributed Computing Systems (ICDCS’10) IEEE, Page No. 98-112.

[13]. Ajeet Mishra, Prof. Umesh Kumar Lilhore, Prof. Nitesh Gupta, "Review of Various Data Storage and Retrieval Method for Cloud Computing", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN: 2456-3307, Volume 2, Issue 5, pp. 584-588, September-October. 2017

[14]. Umesh lilhore and Santosh Kumar, “Analysis of performance factors for cloud computing", International Journal of Information Technology and Management (IJIMTM ignited journal), Vol IX, Issue No. XIV, November 2015, ISSN 2249-4510, PP 305-310.

[15]. Li Chen, Xingming Sun, Zhihua Xia and Qi Liu," An Efficient and Privacy-Preserving Semantic Multi-Keyword Ranked Search over Encrypted Cloud Data", In Proc. International Journal of Security and Its Applications, Vol.8, No.2 (2014), Page No. 323-332.

[16]. R. Sharmila," Secure retrieval of files using homomorphic encryption for cloud computing", In Proc. IJRET: International Journal of Research in Engineering and Technology 2014, Volume: 03 Special Issue: 07, Page No. 845-849.

[17]. Revathy B.D, Anbumani .A, Rohith .V," Enabling Secure and Efficient Multi Keyword Ranked Search over Encrypted Cloud Data", In Proc. International Journal of Science, Engineering and Technology Research (IJSETR), Volume 4, Issue 2, February 2015, ISSN: 2278 – 7798, Page no. 389-394.

[18]. S. Grzonkowski, P. M. Corcoran, and T. Coughlin, "Security analysis of authentication protocols for next-generation mobile and CE cloud services," in Proc. ICCE, Berlin, Germany, 2011, pp. 83-87.

[19]. 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.

[20]. 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.

[21]. Y. C. Chang, and M. Mitzenmacher, "Privacy-preserving keyword searches on remotely encrypted data," in Proc. ACNS, Columbia Univ, New York, NY, 2005, pp. 442-455.