Privacy Preserving Keyword Based Search For Cipher Text Retrieval In Cloud Computing

KASIVISWANADHAM Y [1], DR.CH.D.V.SUBBARAO [2]

[1]Research Scholor,Dept of CSE,, S.V.UNIVERSITY COLLEGE OF ENGINEERING, TIRUPATI, INDIA
[2]Professor, Dept of CSE,, S.V.UNIVERSITY COLLEGE OF ENGINEERING, TIRUPATI, INDIA

[1]ykvnath@gmail.com, [2] subbarao_chdv@hotmail.com

ABSTRACT

Privacy and data security is one of the major challenges in cloud computing. Cloud data owner convert the plain text into cipher text data, before outsourcing it on cloud server (CS). Privacy preserving Public Key Encryptionis to reduce the processing (computational) overhead of data owners while encrypting and decrypting the files without leaking any information about the plain text.A few methodologies have been given to enable searching the cipher text. In this paper, a novel technique to retrieve the cipher text from the cloud server, based on efficient keyword search scheme.The proposed approach allows an authorized user to retrieve matching data based on trapdoor query request. In preprocessing stage, porter stemming algorithm is applied for file collection to extract keywords. Elliptic Curve (EC) key generation is used for generating key for authorized access. Data owner builds a secure, searchable index from extracted keywords using bloom filter. Both files and index are encrypted with blowfish encryption algorithm and stored on cloud server. The authorized user generates trapdoor and sends it to the cloud. Once receiving the trapdoor, CS searches secure index with the corresponding files and matching files are clustered using k-means and each cluster data trained by artificial bee colony (ABC). Finally, the cipher text data are decrypted with blowfish decryption and get the original data. Experiments have been conducted using the collection of documents and the performance are evaluated based on the time cost of computation, communication and search efficiency.

KEYWORDS *—keyword search, cipher text retrieval, cloud computing, porter stemming, blowfish, trapdoor.*

1. INTRODUCTION

Cloud computing (CC) is one of the greatest significant patterns in IT industry in which resources are shared through Internet and the information storage. The computing resources are outsourced to some other user in a ‘cloud’. The outsourced data may possibly have sensitive privacy information. It is needed to encode the data before transmitting it to the cloud servers. The data encryption, however, would result in significant difficulties when users need to get necessary data with search, due to the complications of search over encrypted data. Simply encrypting the data may cause some security issues. So, the sender needs to generate several

keywords based on the data outsourced. Then the keywords are encrypted and stored at the server of the cloud. When the cloud user needs to utilize the data, it can select some relevant keywords and send the CT(ciphertext) of the selected keywords to the data server. Then the CT is used by the server to match the outsourced encrypted keywords, and finally returns the matching results to the end user. CC significantly attracts attention and interest from industry due to the profitability, but it also has challenges including data privacy and confidentiality. The benefits of CC consist of reduced costs and principal expenditures, flexibility, scalability and so on.

Various techniques have been proposed for cipher data retrieval in CC. PaaS is used to provide privacy for storage and processing of customer data in CC. It gives feedback to customer about various privacy techniques applied to data and potential risks associated with it [1]. TPRE(Time based proxy re encryption) which expire the access rights after the specific period of time. Hence, only the users whose access rights are active in the access time can retrieve required data [2]. Generalization techniques are applied to avoid the disclosure of sensitive cloud information by hiding quasi identifier attributes. It is advantage to large data sets by leveraging an index based approach [3]. Encryption and decryption services are separated from the storage service to form a business model for CC. The concept behind this business model is one service provider can operates the encoding and decoding system while the other provider operates the storage and application systems [4]. To encrypt and to decrypt EHRs, CT policy attribute encryption(CPAE) is proposed and they need to possess the set of attributes for proper access. Thus the normal encryption procedures can be replaced as an EHR based cloud systems [5].

Searchable encryption techniques have been created recently to allow users to securely search over encrypted text. In this technique each keyword is assigned with the index and it will search the files relevant to that index. Additional user interaction is required to improve spell check mechanism [6]. A detective, data centric approach is developed to increase security of data and trust in the cloud. Trust Cloud, contains a group of techniques that provides cloud security and accountability at all levels of granularity. This can also applicable in IT systems governing [7]. In HDFS all files are controlled by a central server. The triple encryption scheme, which combines DEA for file encryption and RSA for data key encryption. Then, IDFA is applied to encrypt users RSA private key [8]. The encryptionbased on HA set (HASBE) have been proposed to control the access of data transmitted in CC. Scalability and flexibility can be achieved because of its hierarchical configuration and compound characteristics of ASBE [9]. E-healthcare data manipulation is created in a mobile system using Android OS of Google and S3 cloud service of Amazon. It allows medical imagemanipulation and applications because of its MT (multi touch) technology [10].

To reach confidentiality and privacy in SMEs, three level of security assurance is developed. Based on the data sensitivity, Security management Service(SMS) modules are used for CC. It has a six-layer security model and Security Guidelines layer defines legal and policy constraints. Service level agreements (SLAs) between data receiver and a sender needs to specify quality of

IJSER

service (QoS) requirements based on the security concerns [11].Integrity of data is focused which is based on the concepts of SOA (service oriented architecture) and web services. By using IMS, data can be stored easily and DI can be attained [12]. Remote DI checking is analyzed with PDDS which supports data verification [13]. Analysis of data stored in a cloud based data repository is developed in [14]. Keyword search technique is a searching technique to get the encrypted message in proposed framework. Access control mechanism using PKI was implemented in. It enables secure access to outsourced information and monitor user's access procedures to prevent complications associated with data access [15].

The main concepts of CC are it provides an interface for DS (data storage) and communication between nodes. CDS with DI is assured in cloud data system. Thus the cloud would have to develop its own solutions to verify the CDS server. Hence this mechanism is flexible when the users are revoked [16]. To provide traditional service of computation, a mobile cloud structure named mobi cloud was introduced. It deals with trust management, risk managementand secure routing to enhance communication between data users. Based on the communication and performance metrics of each mobile node, mobicloud supports for MANET operations [17]. Washingtons vanish system (WVS) for self-consuming data at cloud is vulnerable to "sniffer attack" and "hopping attack". To address the problem ofVS, safe vanish is introduced with improved SSSA (Shamir secret sharing algorithm). Here, length of the key ranges is extended to reduce hopping attacks and an improved approach using public key cryptosystem is focused against sniffer attack [18]. By integrating HDFS Pairing andHDFS RSA, hadoop data confidentiality in storage servers can be accomplished [19]. Privacy preserving keyword search scheme is proposed to search over encrypted files without losing information. It reduces client's computational tasks by providing data privacy and user query privacy [20].

IJSER

2. RELATED WORK

Dongxiao Liu *et al*. [21] have proposed kNN (K nearest neighbor) and weighted score techniques to implement searchable encryption over stored data. It is essential to encrypt the storage data to restrict illegal access and modification. The transmitted data was stored in a blind storage which is constructed with the help of CP ABE (ciphertext policy attribute based encryption). EMRS was used for eliminating the risk of sharing keys. Because of very huge collection of documents, EMRS uses sub linear search for efficiency. Keyword dictionary and encrypted index were included to overcome difficulties for accurate search over the cloud storage. A multi-keyword ranked search scheme was developed to enable accurate, efficient and secure search over encrypted data of the cloud.

Encryption based on attributes for secure storage was proposed by Xiang *et al*. [22] to benefit with a multi authorities of CPAB. Privilege data control scheme AC (anonycontrol) was established for privacy of data and to limit access. Server is responsible for storing data and it provides access to the data users. AC distributes the central node to minimize the identity leakage and thus achieves semianonymity. Identity leakage is fully controlled by ACF scheme. It was proved that both AC and ACF are secure with DH (DiffieHellman) bilinear assumption.

Baochun Li *et al*. [23] had proposed a public auditing procedure with revocation of user for integrity. User can easily share and modify data with a group of users in the cloud. Every user needs to compute signature for block of data to ensure data integrity. Once a user is disabled from the server, the data blocks which needs to be re-signed by an existing user. It is possible to access the shared data but it is inefficient due to the size of the data is large. The theory of proxy re-signatures, which allows the cloud to re-signs automatically, so it is not necessary for the existing data users to re-sign. Several auditing tasks were verified to gain group auditing.

ShuminGue*et al*[24]. had proposed RASP data disconcertion and kNNquery techniques for efficient and secure DS storage system. In cloud, it is beneficial to use query data services in terms of scalability. But user does not host some confidential data to the cloud in order to succeed privacy. The work load of data access can be reduced by query processing system. To process with kNN queries, the kNNR algorithm was used with range query RASP algorithm. Several attacks were analyzed to prove that the RASP provides security and efficiency.

The data stored in the cloud may contain confidential information. This data can be encrypted before host it to the cloud. Logical search operations are supported to search over the data using dependent scores and factor of preferences. Top key retrieval schemes for multi key search have been developed by Tom H. Luan *et al*[25]. to encrypt the data index or trapdoor which uses homomorphic encryption procedures. Then the sender generates the secret key and sent to the user through channel. Complex logical search operations were proposed with AND, OR and NO keywords. The traditional sub dictionaries technique provides secure search to attain confidentiality, efficiency and privacy. Hence it allows the user to generate useful data with varying performance. The performance was validated with real world datasets and better performance was derived in terms of functionality and query complexity.

IJSER

3. PROPOSED METHODOLOGY FOR CIPHER TEXT RETRIEVAL IN CLOUD COMPUTING

To develop reliable and efficient cipher text retrieval techniques on large volume of data, a novel technique is proposed in cloud computing. Firstly, a Porter stemming algorithm is to extract keywords in the data pre-processing stage. This proposed work contains two stages which are public key encryption for privacy preserving and keyword retrieval. In the first stage the key for the authorized access can be created using Elliptic Curve Key generation. Then, secure index for each file is created using m-bit bloom filter. Then, blow fish encryption algorithm is used to encrypt the index and files. In the second phase, the trapdoor information is generated. After receiving a trapdoor from the user, the matching files are searched using key word search technique. The cipher text from the data owner are clustered using k-means algorithm and each cluster are trained using ABC technique based up on the index generated from the data owner. A method using artificial bee colony based artificial neural network to update the weights assigned to features by accumulating the knowledge obtained from the user over iterations. Finally the

original files are retained to the data user after decrypting the original files using blow fish decryption algorithm.

Figure 1 shows that, the system classified into three individuals Data Owner, Cloud Server, authorized data user. Data owner consist ofset of files outsource on cloud in encrypted mode. Before outsourcing encrypted files, data owner must generate index for file collection and encrypted it. Authorized user give query request as a trapdoor for searching files. Then the cloud server search the matched files and provide top k-files to the authorized user. Then the user decrypt the matched files and retain the original file.

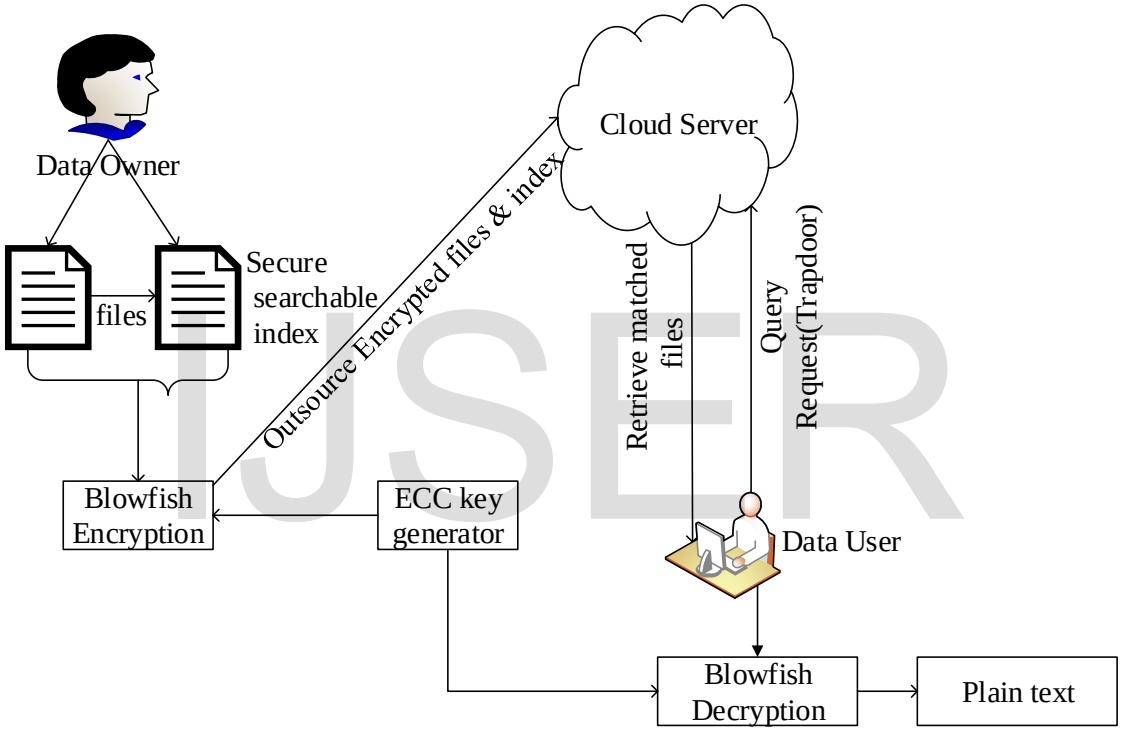

Fig. 1: Architecture of Proposed method

4. PRELIMINARY TECHNIQUES

a) Pre-processing

Data owner (DO) provides huge amount of files $F = \{f_1, f_2, ..., f_n\}$ to outsource on the cloud in cipher text form that he creates searchable index I from a specific keyword set $KW = \{w_1, w_2, ..., w_m\}$ extracted from the file collection F. Porter stemming algorithm is applied to extract keywords from F. Stemming is a basic segment in the preprocessing phase. It is a procedure of etymological standardization, in which the variation types of a word are decreased to a typical frame. It is a standout amongst the most normally utilized truncation stemmers. It removes affixes from a word over various cycles until every one of the principles/conditions are

considered. It is used to speed up the execution. The porter stemming algorithm is described in figure 2.

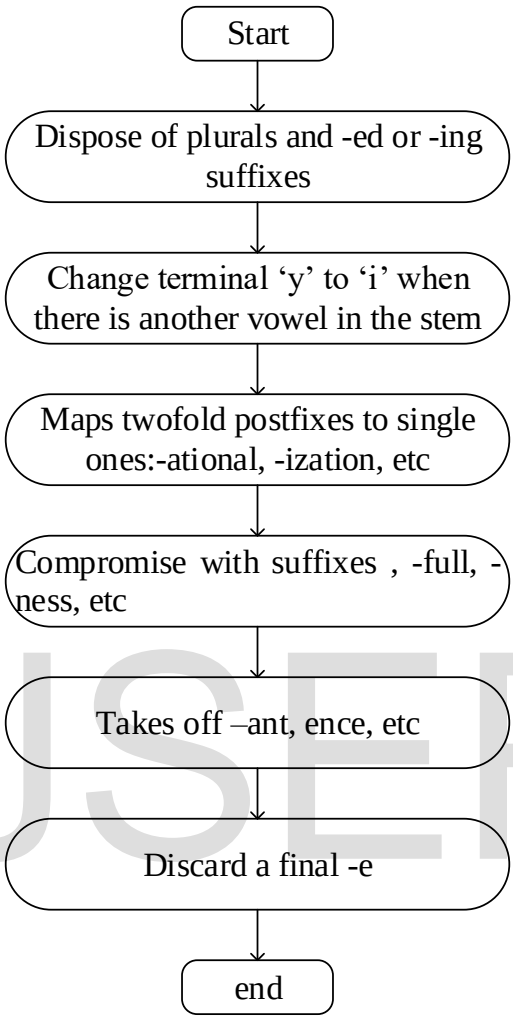

Fig. 2: Stemming Process

Figure 2 consist of six steps and each step perform conditions until one of them passes the conditions. If a condition is accepted, the suffix is detachedproperly, and the further step is executed. The output stem at the end of the sixth step is returned.

b) Public key encryption phase

In this phase data owner generates key for authorized access using elliptic algorithm. Next, blow fish encryption algorithm is applied into index and files.

Step 1: Key generation

Data owner generate the secret key d or public key is used for access the authorized data user. Key generation is an important part where we have to generate both public key and private key. The data owner will be encrypting the message with user’s public key and the user will decrypt

its private key. Every user develop a key pairs to be used for encryption and decryption process. Here Elliptic curve (EC) algorithm is used to obtain keys.

Data user register with data owner to get authentication for data retrieval. Data owner generates key with elliptic curve key generation for user authentication. If the data owner store files to cloud with a secure storage and access control forauthorized user. Elliptic curve groups over real numbers are not practical for cryptography due to slowness of calculations and round-off error. An elliptic curve over a prime field F_p of characteristic greater than three can be formed by choosing the variables g and h within the field F_p and the set of points (x, y) which satisfy the elliptic curve equation is,

$$y^2 = x^3 + gx + h \bmod p \tag{1}$$

Where $x, y \in F_p$ together with a special point ∞ and every value of g *and* h produces a different elliptic curve. The public key is a point in the curve and the random number is a private key. Multiplying the private key with the generator point G in the curve gives the public key.

Let P and Q be two points on an elliptic curve such that,

$$GP = Q \tag{2}$$

WhereQ is the public key, P is the private key and G is the generator point.

If $x^3 + gx + h$ contains no repeated factors, or equivalently if $4g^3 + 27h^2 \equiv 0 (\bmod\ p)$, then these points form a group.

Emulating the geometric construction for addition, the formulas for addition over F_p are given as follows:

Let $P(x_1, y_1)$ and $Q(x_2, y_2)$ be elements of the ECG. Then $P + Q = (x_3, y_3)$, where

$$x_3 = \lambda^2 - x_1 - x_2 \text{ and } y_3 = \lambda(x_1 - x_3) - y_1 \tag{3}$$

λ is determined as,

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} \text{ if } P \neq Q \\ \dfrac{3x_1^2 + a}{2y_1} \text{ if } P = Q \end{cases} \tag{4}$$

Step 2:Secure index generation

The searchable index generated with m-bit bloom filter by data owner. It is a kind of data structure with very high space efficiency. It consists of m-bits and k-hash functions. It makes use

of the m-bit array torepresent a collection, and can determine whether an element belongs to the collection. It is initially set to 0 in all positions and for a given set $KW = \{w_1, w_2, ..., w_m\}$. Bloom filters describe membership information of KW using a bit vector V of length l. For this, k hash functions, $h_1, h_2, ..., h_k$ with $h_i : X \rightarrow \{1..m\}$, are used as described below:The following procedure builds an m bits Bloom filter, corresponding to a set A and using $h_1, h_2, ..., h_k$ hash functions:

Table 1: Secure searchable index generation

Bloom filter algorithm
BloomFilter(set KW, hash_functions, int m) **returns** filter filter = allocate m bits initialized to 0 **foreach** w_i in KW **:** **foreach**hash function h_i : filter[$h_i(w_i)$] = 1 **end foreach** **end foreach** **return** filter

IJSER

False positive rate: We can calculate the false positive rate (f_{pr}) based on the size of the filter, the number of hash functions k and the number of keywords inserted with the formula:

$$f_{pr} = \left(1 - \left(1 - \frac{1}{m}\right)^{kn}\right)^k \approx \left(1 - e^{kn/m}\right)^k \quad (5)$$

The equation (5) is minimized for $k = (m/n) * \ln 2$, then it becomes:

$$f_{pr} = \left(\frac{1}{2}\right)^k = (0.6185)^{m/n} \quad (6)$$

Step 3:Encryption:

After making an index, to guarantee the security of list and documents, the data owner encrypts both file and index. Due to the finite computing power on the data owner side,the files are encrypted by blowfish algorithm. Brute schneier designed this blowfish algorithm and it has a 64-bit block size and a key length of anywhere from 32 bits to 448 bits. There is no attack has been detected to crack the blowfish algorithm. It uses a large number of sub keys. The P-array contains 18 32-bit sub keys. Blowfish is a Feistel network dwelling of 16 rounds and the input is a 64-bit data element, x.

blow fish encryption
Split x into two 32-bit division: xL, xR
For i = 1 to 16:
xL = xL XOR Pi
xR = F(xL) XOR xR
Substitute xL and xR
Next i
Substitute xL and xR (Until the last substitute.)
xR = xR XOR P17
xL = xL XOR P18
Remerge xL and xR

Fig. 3: Encryption Algorithm

In the above algorithm 2, Function F determined is determined below,

Partition xL into 8-bit parts: a, b, c, d

$$f(xL) = ((S_1, a + S_2, b \bmod 2^{32}) XOR S_3, c) + S_4, d \bmod 2^{32} \quad (7)$$

c) Cloud Storage

The cloud storage stores the encrypted file collection and index. After receiving the trapdoor query request from the user, the index searches the corresponding files and send back the related encrypted files to the user.At the point when an authorized userendeavors to inquiry the cloud data, he produces a query request and submits it to the cloud server. Once the request is gotten, the cloud server executes cipher text retrieval based on the index in the cloud and produced matched results using k-means clustering algorithm. At last, the authorized userdecrypts the files retained from the cloud server.

d) Retrieval Phase

In this phase the authorized user retrieve the files from cloud storage using keyword. This phase contains three methods which are trapdoor generation, keyword search and decryption.

Step 1: Trapdoor generation

To search the file collection for a given keyword w, an authorized user generates and submits a search request in a secret form a trapdoor T_w of the keyword w to the cloud server. Hash algorithm can be used for generating trapdoor by data owner. Each keyword associated with the index called trapdoor.Upon receiving the search request T_w, the cloud server isresponsible to search the index I and return the corresponding set of files to the user. The below equation shows that the trapdoor computation is,

$$T_{wi} = \sum_{i=1}^{n} H(w_i)^d \tag{8}$$

Where H is a hash function and d be the random key of user authentication. Then the data user sends trapdoor to the cloud storage. By calling search index, the cloud server locates the matching files of the index. Finally, the cloud server sends back the matched files in a ranked sequence, or sends top-k most relevant files to the user.

Step 2:Search over Encrypted index

First the user needs to receive the secret key from the data owner to search over the encrypted files. For every request from the authorized data user to the cloud storage, the matching files are searched using keyword search technique. It is a kind of searching method that looks down the matching files exhibit in cloud that contain numerous words indicated by the data user. The matching files are clustered by using k-means algorithm. It is one of the most frequently used clustering algorithm. Clustering specifies to the way toward gathering tests into various classes in view of their similitudes.Anyone with public key can write to the data stored on server but only authorized users with private key can search. Public key solutions are usually very computationally expensive however. Furthermore, the keyword privacy could not be protected in the public key setting since server could encrypt any keyword with public key and then use the received trapdoor to evaluate this cipher text.

k-means:To perform clustering of the file collection C are brought together into one data set, D. Then the K-Means clustering technique is applied to perform the clustering of on the whole files. The algorithm starts with a set of data $X = \{f_1, f_2, f_3, ..., f_n\}$ that needs to be clustered into a number of clusters $(k \le n)$. K-means calculates the centers of the k groups, optimizing the error of each group as follows:

$$\min \sum_{j=1}^{n} \sum_{i=1}^{nk} \| f_i^j - C_j \|^2 \tag{9}$$

Where $\| f_i^j - C_j \|^2$ is the distance between a data point f_i^j of the cluster j and the cluster center C_j.This procedure access a set of k initial centers by the below steps:

1. Initial point from the data $X = \{f_1, f_2, f_3, ..., f_n\}$ by a uniform random variable called C_1.

2. For every data f_i, the distance $D(X)$ between f_i and the center C_1 is calculated.

$$D(x) = \sqrt{\sum_{i=1}^{n} (f_i - c_i)^2} \tag{10}$$

3. Then, a new candidate to become center is randomly selected using the probability weighted distribution.

$$\frac{D(x)^2}{\sum_{i=0}^{k} D(x_i)^2} \tag{11}$$

4. Continue steps 2 and 3 until selection of k initial centers.
5. After choosing initial center, k-means algorithm applied.

Here all the matching documents are denoted as points. These points are partitioned into 3 clusters using the k-means algorithm when the k=3. To perform clustering of the file collection C are brought together into one data set, D. Then the K-Means clustering technique is applied to perform the clustering of on the whole files. K-Clusters are generated. The set of clusters $C = \{C_1, C_2, ...C_k\}$ where $C_k (k = 1,2,...k)$ are consisting of group of related documents belonging to a distinct cluster C_i. The documents are clustered using this clustering algorithm and the clustered documents are retrieved accurately from the cluster by ABC algorithm.

Step 3: Training with ABC algorithm

ABC (Artificial Bee Colony) algorithm [26] is applied to identify the optimum solutions and it dwells of three basiccomponents.

1. Food sources: It refers a point of description of optimization problem.
2. Employed Foragers: The number of employed bees is equivalent to the number of food sources. The employed bees store the food source information and share with others according to positive probability.
3. Unemployed Foragers: Their principle assignment is investigating and abusing food source. Two decisions are there for unemployed foragers:
    (i) It becomes an onlooker and decides the nectar measure of food source subsequent towatching the waggle moves of employed bee and select food source as per profitability;
    (ii) It turns into a scout and arbitrarily looks new food sources over the nest.

The inclination of food source by an onlooker bee relies on the nectar amount $V(\theta)$ of that food source. The probability with the food source found at θ_i that will be picked by a honey bee can be showed as,

$$P = \frac{\sum V(\theta_i)}{\sum_{k=1}^{n} V(\theta_k)} \tag{12}$$

From the above equation (9), $V(\theta_i)$ is the nectar amount present at the food source θ_i and n is the number of food sources over the bee. To assess the fitness value, the fitness function will be utilized for the arrangement as opposed to measuring the nectar sum. Its representation is characterized as

$$F_{score} = 2 * \frac{\Pr ecision.\mathrm{Re} call}{\Pr ecision + \mathrm{Re} call} \tag{13}$$

It considers both the precision and the review to register the score. Precision is characterized as,

$$\Pr ecision = \frac{Number\,of\ correct\ results}{Number\,of\ all\ returned\ results} \tag{14}$$

Step 4: Decryption

The matched files are retained from cloud server are send to authorized data user. The files are in cipher text form. The blowfish decryption algorithm is used here to decrypt the file and give the original result. Blowfish is a symmetric algorithm, the same procedure is used for decryption as well as encryption, except that $p_1, p_2, ...p_{18}$ are used in reversed order. The only difference is that the input to the encryption is plain text and for decryption, the input is cipher text.

5. EXPERIMENTAL RESULTS AND ANALYSIS

Performance metrics:This part show a thorough test assessment of the proposed method on gathering of documents. The entire test framework is executed by javaplatform. The execution of the strategy is assessed with respect to theefficiency, Accuracy and securityof search over encrypted data. The time of creating key depends on ellipticcurve does not take more time to produce keys. The execution time of index creation, trapdoor key generation, encoding files, search process and decoding file process are computationally effective. For grouping process, k means calculation is utilized. It gives exact result for grouping the documents. For privacy, most secure blowfish encryption is used.

Table 2: Experimental results of execution time of encryption/decryption, throughput for blowfish and AES algorithm

File size (kb)	Encryption time		Decryption time		Throughput	
	AES	Blowfish	AES	Blowfish	AES	Blowfish
12	7.05	5.62	3	1	1.70	2.13
18	6.5	4.21	5	3	2.76	4.27
20	22.65	16.54	12	10	0.88	1.20

Encryption time:Encryption time is yet another an essential issue since it is fundamentally used to ascertain the throughput of an encryption scheme and it shows its speed. The encryption time

can be characterize as the time that an encryption calculation takes to deliver a cipher text from a plaintext .The throughput of the encryption scheme can be calculated as the aggregate plaintext in bytes encrypted partitioned by the encryption time.

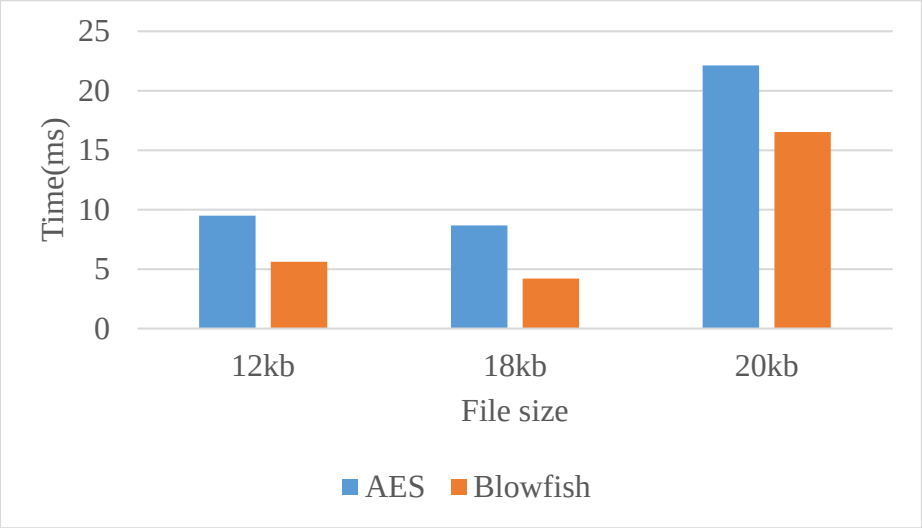

Fig. 4: Comparison on encryption time of AES and blowfish

Decryption time:The decryption time is the inverses of encryption time that can be characterize as the time that a decryption calculation takes to deliver a plaintext from a cipher text.

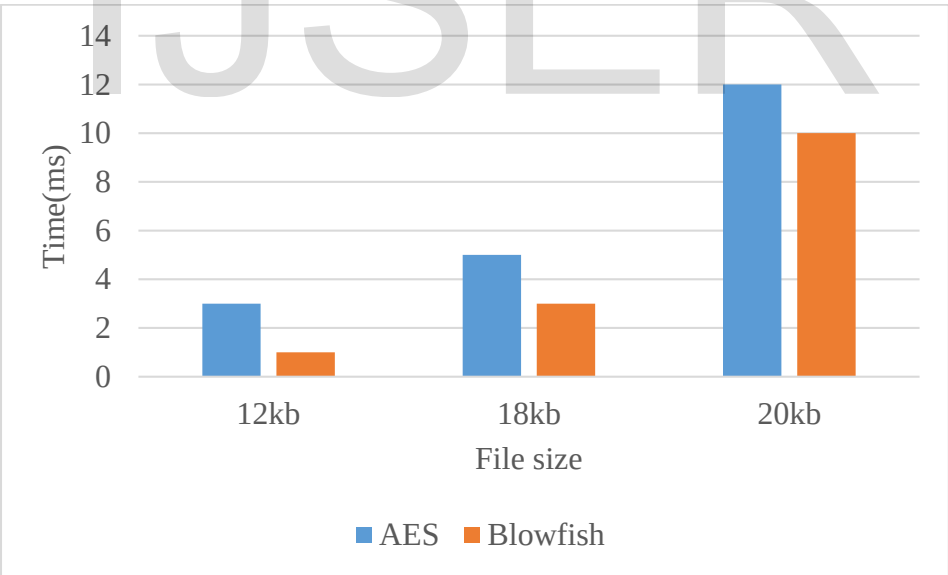

Fig. 5: Comparison on decryption time of AES and blowfish

Throughput:The throughput of the encryption scheme is calculated by dividing the aggregate plaintext in Megabytes encrypted on the aggregate encryption time for every algorithm in.

IJSER

$$Throughput = \frac{File\ size}{Time\ taken\ for\ encryption} \tag{15}$$

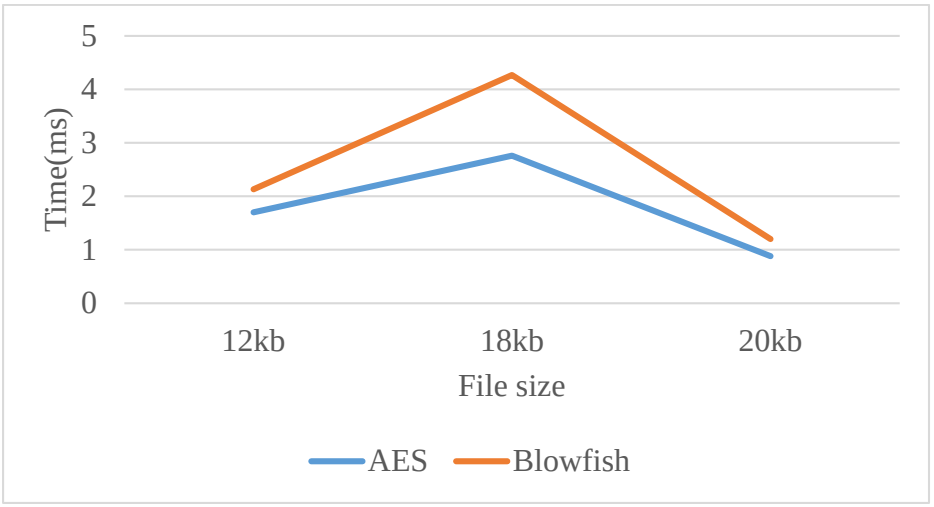

Fig. 6: Comparison on throughput of AES and blowfish

Search precision:

The formula for search precision is,

$$Precision = \frac{True\ Positives}{(True\ Positives + False\ Positives)} \tag{16}$$

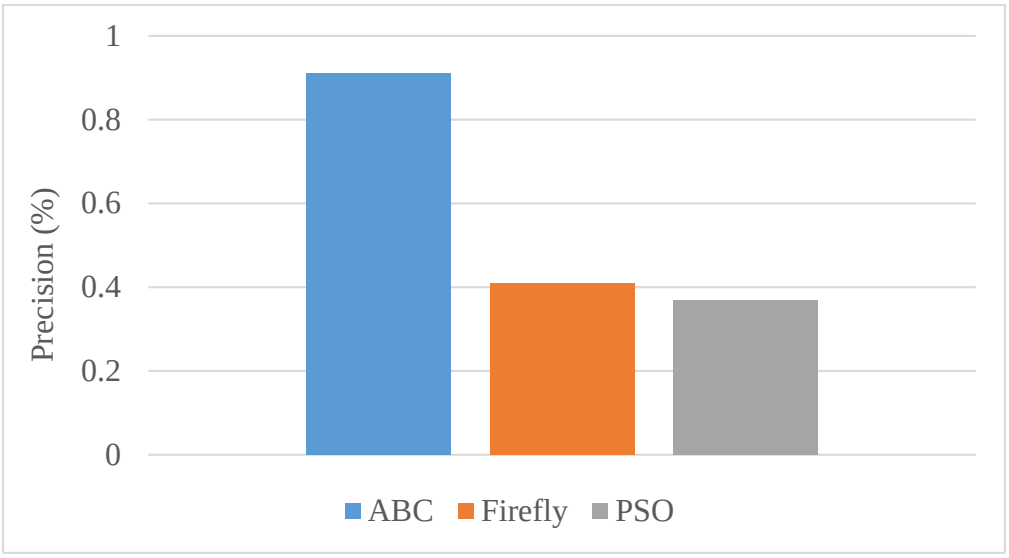

Fig. 7: Comparison on search precision of ABC, Firefly and Pso

Time cost of build index: Whenever a data owner uploads a document into the cloud, keywords are extracted by stemming algorithm. From the stemming words, secure searchable index is

IJSER

generated by bloom filter. Table 4 shows time cost forbuilding secure index for various documents with different number of keywords.

Table 3: Results for index building time

File size	Index building time(ms)	Number of keywords
12kb	3.38	300
18kb	6.2	600
20kb	10.9	700

Table 3 shows the index built time of secure searchable index contains file size and keyword size. The results verifies that proposed work improves the search precision, execution time between retrieved files.

CONCLUSION

This paper concentrate on privacy and secure searching of encryptedcloud information.To search the cipher text content effectively a novel techniquefor searching encrypted data on cloud is proposed.Firstly index for file collection is made and put away both index and file collection in an encrypted format. For indexing and searching two security levels, for example, bloom filter and trapdoor are used. To retrieve the documents, the authorized user generates trapdoor and send it to the cloud. Then the cloud searches the corresponding files over the encrypted data by trapdoor.Then cloud server returns the matching files back to the user. Bloom Filter has been used to make the keyword more secure and secret. The assessment of experimental outcomesshows that searching encrypted data on Cloud is secure and defensive from unapproved client. The technique is proposed for cloud environment where a lot of information is put away in encrypted form. Finally experiments have been handled using the accumulation of documents and the performance are evaluated based on the privacy of owner, time cost of computation, communication and search of cipher text retrieval.

IJSER

REFERENCE

[1]MaristellaRibas, Alberto Sampaio Lima, Jose Neuman de Souza, Flavio Rubens de Carvalho Sousa, and Leonardo Oliveira Moreira, "A Platform as a Service Billing Model for Cloud Computing Management Approaches ", *IEEE, Latin America Transactions*, Vol. 14, no. 1, pp. 267-280, 2016.

[2] Qin Liu, Guojun Wang, and Jie Wu, "Time-based proxy re-encryption scheme for secure data sharing in a cloud environment",Elsevier, *Information Sciences*, Vol. 258, pp. 355-370, 2014.

[3] Xuyun Zhang, Chang Liu, Surya Nepal, Suraj Pandey, and Jinjun Chen, "A privacy leakage upper bound constraint-based approach for cost-effective privacy preserving of intermediate data sets in cloud", *IEEE Transactions on Parallel and Distributed Systems* Vol. 24, no. 6, pp. 1192-1202, 2013.

[4] DimitriosZissis and DimitriosLekkas, "Addressing cloud computing security issues",Elsevier, *Future Generation computer systems*, Vol. 28, no. 3, pp. 583-592, 2012.

[5] Ming Li, Shucheng Yu, Yao Zheng, Kui Ren, and Wenjing Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption", *IEEE transactions on parallel and distributed systems*, Vol.24, no. 1,pp. 131-143, 2013.

[6] Cong Wang, Cong, Ning Cao, Kui Ren, and Wenjing Lou, "Enabling secure and efficient ranked keyword search over outsourced cloud data", *IEEE Transactions on parallel and distributed systems*, Vol.23, no. 8, pp.1467-1479, 2012.

[7] Lijun Dong, Kui Wu, and Guoming Tang, "A Data-Centric Approach to Quality Estimation of Role Mining Results", IEEE *Transactions on Information Forensics and Security*, Vol.11, no. 12,pp. 678-2692, 2016.

[8] Chris J Mitchell, "On the security of 2-key triple DES",IEEE Transactions on Information Theory, Vol. 62, no.11, pp. 6260 – 6267, 2016.

[9] Zhiguo Wan,Jun'E. Liu, and Robert H. Deng, "HASBE: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing ", *IEEE transactions on information forensics and security*, Vol. 7, no.2, pp. 743-754, 2012.

[10] Emma M Mercier and Steven E. Higgins, "Collaborative learning with multi-touch technology: Developing adaptive expertise", Elsevier, *Learning and Instruction*, Vol. 25,pp. 13-23, 2013.

[11] Jesus Luna,, Neeraj Suri, Michaela Iorga, and Anil Karmel, "Leveraging the Potential of Cloud Security Service-Level Agreements through Standards" *IEEE Cloud Computing*, Vol. 2, no. 3, pp. 32-40, 2015.

[12] Yan Zhu, Hongxin Hu, Gail-JoonAhn, and Stephen S. Yau, "Efficient audit service outsourcing for data integrity in clouds ", Elsevier, *Journal of Systems and Software* , Vol. 85, no. 5, pp. 1083-1095, 2012.

[13] Tiago Marques Godinho, Carlos Viana-Ferreira, Luís A. Bastião Silva, and Carlos Costa, "A Routing Mechanism for Cloud Outsourcing<? brk?> of Medical Imaging Repositories", *IEEE journal of biomedical and health informatics*, Vol. 20, no. 1 , pp. 367-375, 2016.

[14] Yan-Bo Han, Jun-Yi Sun, Gui-Ling Wang, and Hou-Fu Li, "A cloud-based BPM architecture with user-end distribution of non-compute-intensive activities and sensitive data",Springer, *Journal of Computer Science and Technology* , Vol. 25, no. 6, pp. 1157-1167, 2010.

[15] DionysisKefallinos, and EfstathiosSykas , "A public key infrastructure model for privacy-enhancing general purpose eIDs" , *IET Information Security*, Vol. 9, no. 2, pp. 91-99, 2015.

IJSER

[16] Yuan Zhang, Chunxiang Xu, Hongwei Li, and Xiaohui Liang, "Cryptographic Public Verification of Data Integrity for Cloud Storage Systems", IEEE *Cloud Computing, Vol.* 3, no. 5, pp. 44-52, 2016.

[17] Jun Wu, Jian Wu, Hao Cui, Chong Luo, Xiaoyan Sun, and Feng Wu, "DAC-Mobi: Data-Assisted Communications of Mobile Images with Cloud Computing Support ", *IEEE Transactions on Multimedia*, Vol.18, no. 5, pp. 893-904, 2016.

[18] Zhen Wang, Mark Karpovsky, and Lake Bu, "Design of Reliable and Secure Devices Realizing Shamir's Secret Sharin",IEEE Transactions on Computers, Vol. 65, no.8, pp. 2443 – 2455, 2016.

[19] IvaniltonPolato, ReginaldoRé, Alfredo Goldman, and Fabio Kon, "A comprehensive view of Hadoop research—A systematic literature review",Elsevier, *Journal of Network and Computer Applications*, Vol. *46, pp.* 1-25, 2014.

[20] Ning Cao, Cong Wang, Ming Li, Kui Ren, and Wenjing Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data", *IEEE Transactions on parallel and distributed systems*, Vol. 25, no. 1,pp. 222-233, 2014.

[21]Hongwei Li, Dongxiao Liu, Yuanshun Dai, Tom H. Luan, and Xuemin Sherman Shen, "Enabling efficient multi-keyword ranked search over encrypted mobile cloud data through blind storage", *IEEE Transactions on Emerging Topics in Computing*, Vol. 3, no. 1, pp. 127-138, 2015.

[22]Taeho Jung, Xiang-Yang Li, Zhiguo Wan, and Meng Wan, "Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption", *IEEE Transactions on Information Forensics and Security*, Vol. 10, no. 1, pp. 190-199, 2015.

[23] Boyang Wang, Baochun Li, and Hui Li, "Panda: public auditing for shared data with efficient user revocation in the cloud", IEEE *Transactions on services computing*, Vol. 8, no.1, pp. 92-106, 2015.

[24]Huiqi Xu, ShuminGuo, and Keke Chen, "Building confidential and efficient query services in the cloud with rasp data perturbation" *IEEE Transactions on Knowledge and Data Engineering*, Vol. 26, no.2, pp. 322-335, 2014.

[25] HongweiLi, Yi Yang, Tom H. Luan, Xiaohui Liang, Liang Zhou, and Xuemin Sherman Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data", *IEEE Transactions on Dependable and Secure Computing*, Vol. 13, no. 3, pp. 312-325, 2016.

[26] Karaboga, Dervis, and BahriyeBasturk, "On the performance of artificial bee colony (ABC) algorithm", Elsevier, *Applied soft computing*, volume. 8, no. 1, pp. 687-697, 2008.

IJSER