Volume 66, Issue 2, 2022

Journal of Scientific Research

of

The Banaras Hindu University

Efficient and Secure Multi-Keyword Ranked Search and Group Data Sharing for Encrypted Cloud Data

Snehal N. Sarode[1] [0000-1111-2222-3333] and Geetha R. Chillarge[2] [1111-2222-3333-4444]

[1, 2] Department of Computer Engineering Marathwada Mitra Mandal's College of Engineering Pune, India
[1]sarodesnehal10@gmail.com
[2]geethasb@mmcoe.edu.in

Abstract. Nowadays due to the popularity of clouds and their quality services, many of the users such as organizations, industries as well as individuals are migrating towards the cloud to store their important, confidential data and get easy access to their data anywhere anytime over the internet. Different encryption techniques are employed in cloud computing to ensure data confidentiality, security, and privacy. This makes it harder for end-users to retrieve precise data. Due to the huge amount of data over the cloud and multiple data users, secure data storage and retrieval are required. So in this paper develop an efficient data group sharing and multi-keyword ranked search method for encrypted cloud data collection in this research work. The developed system is implemented using the El-Gamal cryptography algorithm to provide security through effective key generation techniques and encryption strategy. Here, a multi-owner data setting is used instead of a centralized data owner setting; each member of the system in one particular group gets equal rights for both searching and sharing functionality and this may increase system usability. By taking into consideration lots of data in the cloud, the vector space model and TF-IDF model are utilized and according to the cosine similarity score, the method generates a ranked multi-keyword search result to deliver effective query result from numerous data and enhance secrecy in the situation of numerous data owners. In this system searching efficiency is improved by developing an index-based search structure. In the group, data can disperse with co-owners/users by developing a role-based access policy (RBAP), and a user revocation strategy is developed with low computation time as well as communication overhead. At last, the efficiency and security of a developed system are exhibited by broad exploratory assessment.

Keywords: Cloud Computing, Searchable encryption, Searching, Data Sharing, Security, Privacy, EL-Gamal, Multi-Keyword, Search Query, Public Key Encryption, Multi-Owner.

1 Introduction

In the present advanced world, cloud computing gives many advantages to their customers or user like good geographical scope in the shortest period, mobile accessibility, great framework on a low financial plan, and so forth. (13). In fact, there are numerous cloud applications that enable either group data sharing or multi-keyword ranked search in cloud environments both having lately been a popular topic, but both the application developed in one scheme did not happen earlier. It is critical to address the problem of achieving both safe group data sharing and multi-keyword ranked search on encoded cloud data in one scheme by considering cloud computing needs. Data is kept in logical pools as digital data in cloud storage. The same data will have several owners in a multi-owner situation. There will be a primary server to manage all of the data. Multiple servers may be present in the cloud, and they may be located in different places. The security and management of the stored data will be the responsibility of the primary server or cloud storage providers. These cloud storage companies will sell or lease storage space to cloud customers. Cloud storage provides network access to digital data that is dispersed and scalable. The safe search over encrypted data is an issue that must be addressed in cloud storage. Secure search on encrypted cloud data is the most difficult job in cloud storage. There are a variety of search methods available. However, they either cause system overhead or make such techniques very difficult to implement across huge data sets. Data will be kept in the cloud in encrypted form to prevent unauthenticated access. An index-based multi-keyword ranked search strategy is built to offer an efficient search [1]. An index is created by identifying the words that seem to be keywords in a text. After then, all of the indexes that have been created are combined into one. The top results are returned using the TF-IDF model [3].

If a user wants to get just documents that include certain terms, he or she must specify any type of word-to-document mapping using the appropriate keywords. The user must first establish the mapping or any technique to the cloud storage for effective data retrieval. The technique must function without jeopardizing data security. Through the allocated space in a cloud, a user may read and write data over the internet. This file-sharing may take place from anywhere. Because all activities are performed on the server, a reliable backup and recovery solution is required. However, if data storage is done without appropriate security procedures, there are numerous security concerns. There are many third-party suppliers available nowadays. These vary in the security measures they have implemented. There are many papers that may have multiple owners. However owing to security precautions, not all of the data will be accessible to all of the owners. The health record system of some patients in the health care industry is an example of this. Patients, like data users, must have access to records about their health problems. They must do it by gaining access to top data files from various data proprietors. In a personal health record system, a data user, such as a patient, should be able to receive their top documents

DOI: 10.37398/JSR.2022.660210

towards a certain particular instance from several data owners. Health monitors, hospitals, physicians, and other data owners may be among them [5]. Workers in an organization should also be able to search document files created by different employees. The top-k query for numerous data owners, in which the cloud server may combine various data indexes encrypted with separate keys and enable top-k query effectively. Fostering an effective system for many data owners, as opposed to the single-user situation, becomes a problem. An index structure for every data owner's encrypted files may be created to enable privacy preservation and speedy searches. Data users must create a trapdoor for every data owner for a particular query condition, and the cloud ought to likewise look through each index. Because of the direct association between the quantity of trapdoors and data owners, this is clearly inefficient. Allowing every data owner to use a similar key to encrypt their data files is a straightforward method to get around this restriction. Nonetheless, if one of the owners is compromised, the system may crash [10].

2 Literature Survey

In this part, we present an overview of existing literature on different schemes. For cloud security, a new method combining ciphertext policy-identity attribute-based encryption (CP-IDABE) with the rivest-shamir-adelman (RSA) algorithm is suggested [1]. The public and separate secret keys produced by the automated certificate authority (ACA) are given from both the owners and users. The property strategy distinguishes between the user and the owner when it comes to cloud data access. The suggested RSA-CP-IDABE method also successfully protects against the man in the middle (MITM) attack. For the mysterious strategy so over-encoded information under their separate access strategy, the CP-IDABE incorporates that both characteristics and the ID of the user/owner. The unsymmetrical security key is used by the clients to receive data in the cloud. In addition, the secondary owners of data were given their secret key in the present scheme. The user/owner creates a username and a password that will be used to identify themselves in the proposed system. The attribute set that determines whether an individual is a client or a proprietor is defined by the property set that encases the advantages of access. Furthermore, while accessing the cloud via the ACA, a series of responses to a group of questions [2] is utilized to verify the client or proprietor. Huang et al. [3] suggested a new cloud-based method for securing data sharing among users and establishing controlled dissemination for various owners. To transfer data among users acquired from its proprietor, the identity-based broadcast encryption (IBBE) method is used. Additionally, the owner specifies a fine-grained access structure depending on the preferences. The suggested method has been shown to offer sufficient data security in multi-owner clouds.

In a cloud with a multi-owner context, Miao et al. [4] proposed a private information method built using attribute-based keyword search algorithms. The suggested system made it easier to track down fraudulent users. In encrypted form, the method is helpful for providing sufficient security and preventing the keyword guesses attack. On data sets, the suggested scheme's performance is evaluated. To secure the cross-cloud environment, Anand et al. [5] developed an ECC-based diffie-hellman key- exchange protocol and digital signature. The suggested method protects multi-owner data integrity. Furthermore, it ignores the user's perspective, which includes a variety of characteristics. Effective and security safeguarding multi-keyword graded search scheme with fine-grained access control (MRSF) [6]. MRSF may conduct extremely exact ciphertext retrieval by integrating coordinate matching with term frequency-inverse document frequency (TF-IDF) and enhancing the safe KNN procedure. Moreover, it can successfully refine client search advantages by utilizing the polynomial-based access method. A formal wellbeing study shows that MRSF is secure as far as the confidentiality of outsourced data and the protection of records and tokens. Furthermore, thorough investigations indicate that, when compared to existing methods, MRSE provides higher search accuracy and more functionality effectively. States that retrieving needed records from the encrypted cloud turns into a challenge that necessitates looking through the encrypted data [7]. Utilizing the information structure bunch B+ tree, a productive multi-keyword ranked search method over encrypted data in the cloud. They Fabricate a B+ tree file structure dependent on a collection of data sets to enhance query efficiency. This can advance the index structure and deal with productive and quick pertinence between the query and cloud data. We apply the enhanced KNN-based algorithm to encode delicate data, specifically for the privacy issue of query data; the searchable encryption of this method accomplishes exact multi-keyword queries across encrypted cloud data and provides the most significant top-k outcomes. Broad test discoveries on true informational collections show that the suggested method may decrease index storage and increase retrieval performance considerably.

As indicated by kaiping Xue [8] another heterogeneous design is proposed to address the single-point execution bottleneck issue while also providing another more robust access control framework with a reviewing component. In our framework, a few attribute authorities are utilized to spread the load of user validity confirmation. Nonetheless, in our framework, a CA (central authority) is utilized to create concealed keys for clients whose authenticity has been verified. Unlike previous multi-authority access control systems, ours manages each authority's full attribute collection separately. To upgrade security, we additionally propose an examining technique to distinguish whether the AA (attribute authority) has played out the validity confirmation process incorrectly or maliciously. Kan yang and Xiaohua Jia [9] presented a revocable multi-authority CP-ABE system and utilized it to assemble the basic components of the information access control scheme. With our attribute revocation tool, you can easily accomplish both forward and backward protection. The framework typically designs an expressive, reliable, and revocable information access control instrument in a multi-authority cloud storage framework, where several authorities coexist and every authority might give credits freely. The system [10] presented a safe approach for anti-collusion key distribution that does not rely on third-party networks and allows users to get their secret data securely from the group owner. Secondly, this strategy allows for fine-grained access control; any member of the community may access the cloud source, and revoked users are unable to re-access the cloud after they have been revoked. Third, the method will defend the scheme against collusion assaults, ensuring that revoked clients cannot get to the genuine information record even if they combine with an untrusted cloud. Using polynomial capability, the system may complete a safe client negation conspiracy in this manner; lastly, this plan can achieve excellent performance, indicating that previous clients do not need to renew their revocation from the community.

According to [11] it is suggested that the important feature's most notable characteristic is that it is centered on KP-ABE, which includes non-monotonic access structures and standard ciphertext sizes. The company also provides the first key-policy attribute-based encryption (KPABE) technique with a fixed ciphertext size that permits non-allowed access structures. To do so, the framework first shows that a specific class of character-based transmission encryption strategies produces a monotonic KPABE framework in the chosen set model. After that, the framework provides a novel identity-based revocation system that, when joined with a specific instance of our generic monotonic design, gives the very first genuinely expressive KP-ABE implementation with the constant-size ciphertext. F. Zhang and K.Kim [12] suggested an alternative. Both approaches are based on ID-based ring signatures and are focused on bilinear pairings and the java pairing library. Furthermore, the report detects their security and performance to other current methods. The java pairing library (JPBC) was utilized for information encryption and decryption. Many dynamic access control rules are intended for end customers but simultaneously safeguard the privacy and confidentiality of the data owner.

The existing study revealed that many plans are planned by utilizing diverse encryption algorithms for accomplishing either data searching or sharing operations over the cloud, but very few of them perform two tasks safely, likewise, the greater part of the plans which back search activity just helps a single keyword search. In single–keyword searching, whatever user searches, it gives output as the list for exactly one term, it cannot give a variation of that keyword and that's why it is a time-consuming process. Many of the existing systems contain revocation options, but it contains lots of computation overhead of key updating at each time of the revocation of a group member.

Our contribution is summed up as follow:

1. Secure plans such that they support multi-keyword-based rank searching along with performing sharing operations for group users in a single scheme. Again instead of linear searching, developing an index-based search structure with a rank search result increases the efficiency of the system.
2. In the group data sharing multi-owner data, the setting is developing instead of a centralized data owner setting; each member of the system in one particular group gets equal rights which may increase system usability.
3. The system is implemented using the El-Gamal cryptography algorithm to provide security through effective key generation techniques and encryption strategy. The system adopt two threat models i.e. known ciphertext model and known background model that enhanced system privacy from internal threat i.e. insider attack as well as an external threat i.e. man-in-the-middle attack, chosen plaintext attack, collision attack, etc.
4. Data can disperse by developing a role-based access policy and user revocation strategy is adopted with low computation time as well as communication overhead.

3 Proposed system

3.1 Design Objective

The principal objective of the proposed system is to incorporate group data sharing and multi-keyword ranked searching over outsourced data over the cloud. To achieve this main principle our system design focus on access control, data confidentiality, security, and privacy, multi-keyword ranked search mechanism, and efficiency as follow:

- Access control: To accomplish access control inside to provide authorized access to data files. Here two types of access control are provided---First any member within only that particular group can securely perform operations and utilize cloud resources. Second is after revocations of any group members as well as any unauthorized users are not able to perform any operation and access the cloud data. A role-based access policy (RBAP) is adopted to disperse data with authorized co-owners/users
- Data confidentiality, security, and privacy: To prevent unauthorized users including cloud servers from knowing the content of stored data files as well as index information and maintain privacy and security. Maintaining the availability of confidentiality and security is the most important and challenging task. The El-Gamal cryptography algorithm is utilized to provide security through effective key generation techniques and encryption strategy. Since the keys are truly challenging to expect, this is viewed as a productive strategy for encryption and decoding. The main reason behind adding a mark to an informing exchange is to ensure it against MITM, which our strategy may do very well. Again two threat models i.e. known ciphertext model and known background model have enhanced system privacy from internal i.e. insider attack as well as an external threat i.e. man-in-the-middle attack, chosen plaintext attack, collision attack, etc. Again after the revocation of any member to achieve data confidentiality and data security they are not able to access his data and implement a database security approach also.
- Multi–keyword-based rank searching mechanism: To configuration search plots that permit multi-keyword queries and give effective data retrieval ranked strategy. An index-based searching strategy is developed to improve the efficiency of searching in terms of the time required for searching. The index is constructed using vector space model strategy and TF-IDF technique by weighting the keywords in each document and getting encrypted. In the present system for multi-keyword searching concepts "co-ordinate," matching principle is developed. The ranking result is provided by calculating the cosine similarity score between the query and document index and delivering effective query results from numerous data.
- Efficiency: To achieve all mentioned objectives on functionality and security must be done by low computation and communication overhead. The searching efficiency is adopted by utilizing the concept of an index-based searching strategy. The efficiency of the overall system is exhibited by broad exploratory assessment with existing one.

3.2 Motivation

Encryption is usually the main strategy to protect sensitive information when uploading data to the cloud. At the time of encrypting the data, search structures were no longer active, since the query could not look at over the encoded data. It is challenging to perform searching and sharing operations over the ciphertext data. Also in cloud-based group sharing security is the main issue. Sometimes there is a possibility of personal data leakage while handling it among multiple users is the most important issue. So to maintain data security each data file is stored by its own keys and after being denied of any member they are not able to access his data. Also, it is required to restrict the malicious members from accessing the data in group file upload constraints will be introduced. When uploading data over the cloud, it is necessary to maintain confidentiality using encryption.

3.3 System Model

In this research, we propose an efficient multi-keyword ranked search and a secure sharing scheme with multiple data owner settings. It suggests that any client within the group can search as well as share data with others within the group through an untrusted cloud. Specifically, it utilizes cryptographic natives to ensure information and offer documents. Data confidentiality will be maintained using encryption when uploading data to the cloud, EL-Gamal public key cryptography algorithm is used to generate keys for each data file which are stored by the owner over the cloud [16]. In Cloud-based group data sharing to restrict the malicious members from accessing the data in group file upload constraints will be introduced by the group name as well as a group member and achieve access policy. In the process of efficient data retrieval, here use multi-keyword ranked search. It provides the search result to the end-user as the most relevant document is ranked form as an output. The index-based searching process is established to maintain the efficiency of searching. The construction of index and query generation is taking place by using the combination of both vector space model and term frequency (TF) and inverse document frequency (IDF) [6] techniques.

In the system model, multiple data owner setting is there so data owner is that the group manager or the group members (anybody within the group) of any industry/organization holds a collection of files that they want to share in the group. Before uploading a file the keywords set are extracted and will construct/build the searchable index as per the keyword extracted from the document assortment. Here For pre-processing and keyword extraction avoid common words, stop words, punctuation, converting to lowercase, punctuation, white space, etc. these methods and algorithms are used. After preprocessing step apply the TF-IDF algorithm to weight the terms in each document vector then both index and file collection gets encrypted using the public key generated by the El-Gamal cryptography algorithm and will transfer ciphertexts to the cloud server. To share a particular data files with group member's data owner applied a role-based access policy (RBAP) mechanism and according to that only that particular user can access it and decrypt that particular file.

Authorized users of that particular group can enter any interesting keywords as query keywords and these keywords are encrypted and afterward send as a search request to CSP to build the vector of the query. The cloud server receives keywords set as encrypted query keywords and starts searching the index. Here cosine similarity measures are applied over the frequency and will return the set of match documents in encrypted form. As per the similarity score, documents are displayed in a ranked form. After receiving the document, the group member has to take permission of the data owner to decrypt and download that particular file, here again, access control is provided. Subsequently, unapproved users, as well as storage servers, can't get easy access to the content of the data files. All group members of a particular group have the rights to share and retrieve data as far as they belong to that group. The group manager removes group members when they leave the group or move to a different group. Members are also revoked due to their unauthorized action and after the revocation; they are not able to access his/her data as their data is permanently blocked. In the future, if revoked data owner tries to access the data it is impossible to directly connect.

3.4 Threat Model

Whenever we are going with the clouds, we are considering that the security constraint. But the cloud is trusted but curious about the data that we are putting over the clouds. For providing better services the clouds may be curious about the data files or for maintaining certain sets of profile that they want to know like what kind of that data that is or what kind of data set the user want to access and searchable index available over the server. According to the data available over the cloud in a proposed system we adopt two threat models.

The first one is known as the ciphertext model. It is the model in which cloud server contains all types of encrypted information i.e. encrypted data files, encrypted index, and encrypted query keywords. To protect all the data files, indexes, and queries from cloud server threats like a man-in-the-middle attack, chosen-plaintext attack, collision attack, insider attack, and use to build a searchable index.

The second one is a known background model. Cloud server contains important information as compared to the first model. It is one of the strongest models in terms of knowledge. This model contains all the statistical information related to data files, keywords such as information of term frequency of all the terms, and inverse document frequency. The cloud server may launch statistical attacks over the statistical information available in this model to identify the query keyword.

4 Preliminaries

4.1 TF*IDF Rule and Vector Space Model:

For similarity retrieval over plaintext, many different strategies are there among them vector space model is a widely used method for multi-keyword ranked search. Vector space model is the algebraic model that represents each document in n-dimensional space of vector. The keywords in the document along with its term frequency weights are the elements of each document vector. Similarly, the query keywords are also represented in an n-dimensional query vector along with its inverse document frequency weights. The TF*IDF rule plays an important role to provide accurate similarity results [7]. In TF*IDF Rule, TF implies Term Frequency it estimates how many times the term appears in a document file and IDF implies inverse document frequency calculate the recurrence of a specific term in the archive file corresponds to the total number of documents file. This strategy is useful to fulfill the objective of the multi-keyword ranked search.

Terminology:

t – Term (word)

d – document (set of words/terms)

N – Count of the corpus (total set of documents)

D_d – Document vector

QV **–** Query vector

$TF_{t,d}$ – Term frequency of term t within the document d

$tf_{t,d}$ – Term frequency of term t within the document d

$IDF_{t,N}$ – Inverse document frequency of term t within the corpus N

df_t - Count of the document containing the term t

W_t – TF-IDF weight of term t

$|D_d||QV|$ – Product of the magnitude of vector terms in D_d *and* QV vectors.

i – Number of documents range from 1 to n.

The term frequency (TF):

$$TF_{t,d} = \frac{count\ of\ t\ in\ d}{number\ of\ words\ in\ d} = (1 + \log tf_{t,d}) \qquad (1)$$

The inverse document frequency (IDF):

$$IDF_{t,N} = \log\left(\frac{N}{df_t}\right) \qquad (2)$$

$TF * IDF$ score calculated by using the formula:

$$W_t = TF * IDF = (1 + \log tf_{t,d}) \, .\log\left(\frac{N}{df_t}\right) \qquad (3)$$

To measure similarity between document vector D_d and query vector QV cosine similarity measures are applied. Similarity measure score is calculated by using inner product between the two equal length vectors i.e. document vector D_d and query vector QV. The resulting score of similarity always ranges between 0 and 1. The higher score 1 contains high relevancy between the query and document vector, vice versa and it is calculated as:

$$Similarity\ score(D_d, QV) = \frac{D_d \times QV}{|D_d||QV|} = \frac{\sum_{i=1}^{n} D_{di} \times QV_i}{\sqrt{\sum_{i=1}^{n}(D_{di})^2}\ \sqrt{\sum_{i=1}^{n}(QV_i)^2}} \qquad (4)$$

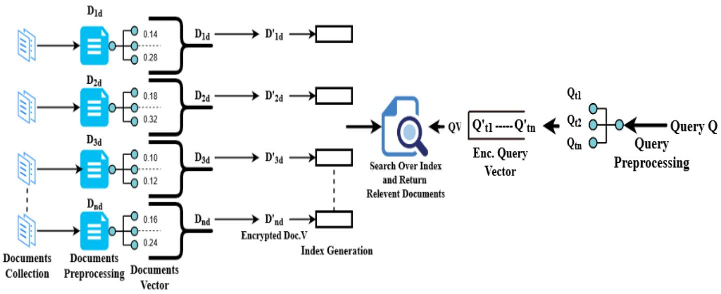


Fig. 1. Overview of index construction

4.2 Secure Index Construction:

In the proposed scheme we adopt a searchable index-based multi-keyword-based ranked search. In the process of index construction the whole document collection D_d vector is divided into sub-vector D_{id} which contains the set of keywords of that particular D_{id} and its weights. Similar to the document vector, query vector QV is also divided into sub-vectors QV_i for ith document sub-vectors. To calculate the final similarity measure, at last, the average of all sub-vector scores is calculated.

In this way, the cloud server gives the relevant document to the end-user for the particular query. In the process of index construction, TF*IDF plays an important role; according to the weight score the document vector is generated. Fig. 2. gives the overview of index construction.

5 System Architecture

The problem that we consider is the secure data sharing and multi-keyword ranked search. Over the private database model where data files are encrypted and are holed by the cloud server. The system architecture of this proposed scheme is given in Fig. 2.

It gives an overview of the whole process. The framework comprises four significant elements: admin, group manager, group members, and cloud service provider. The functionality between the modules of the framework is described by four functional modules: data file encryption module, data file decryption module, rank-search module, and revocation module.

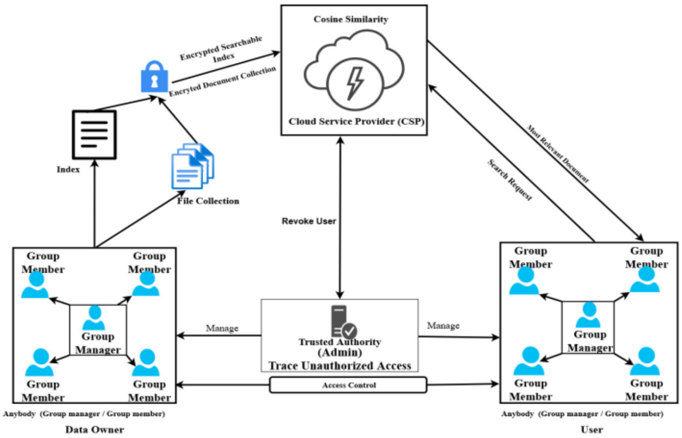


Fig. 2. Proposed system architecture

5.1 Modules:

1. Admin Module: Admin is a fully trusted third party for the overall system. In this module, the admin can view each and every action performed in the system by the group manager as well as group members of every group and trace that action. Admin contains the revocation request list of group members sent by the group manager. Admin has the right to revoke group members permanently from the group after that they are not able to log in to their account to access his/her data.
2. Group Manager Module: The group manager is the head, for the generation of the group by a unique name, group boundaries, managing group members (i.e., authorizing group members.), group member registration, group member revocation. Group managers know the information of that particular group only (means group member credentials). The group manager in our plan is a completely confided third party to both the cloud and group members. They can perform both searching and sharing operations according to their role (data owner/user). There is one restriction that every group contains only one group manager. If the group manager wants to revoke a particular group member/user of that group it can send the request to the admin.
3. Group Member Module: Group members are a bunch of intrigued users registered to a particular group in the proposed plot; members are people with interests (e.g., specialists, money managers, brokers, and so forth). In this module, the group members select the particular group when they do registration hence they get identity with the group name from which they register also to provide access control this identity is important. Everyone Group Member within the group can perform both data sharing as well as search operations according to their role (data owner/user) due to the multi-owner setting. Group members get registered into the particular group with the permission of the group manager.
4. Cloud Service Provider (CSP): The cloud service provider (CSP) provides unlimited storage space and services to the user. Here all the uploaded data and searchable index are securely stored. Cloud can perform searching and sharing operations according to the command provided. However, the cloud has the characteristics of honesty but curiosity. They are straightforward yet fairly inquisitive. In different words, the cloud won't purposely erase or adjust the transferred information of users; however, it will be interesting to comprehend the contents of the put-away data and the user's identity.

For better understanding, if the proposed system architecture is applied to college as one organization then admin acts as a college principal, as fully trusted third party for the overall college, they know each and every action performed in every department of college also they have rights to take proper action (revocation) accordingly. Now, each department is nothing but a group and here the group manager of the department is HOD who has deep knowledge of that particular department and completely confided the third party. They can take a decision at the department level and lastly take the final call from the principal, just like group manager in the proposed system do to revoke group member. Lastly, all the teachers in a particular department are acting as group members. Just like HOD and teacher both can communicate, discuss, share information with each other, both group manager and group members can perform both searching and sharing operations.

5.2 Functional modules

1. Data file encryption module. Using this module every data file uploaded over the cloud by the data owner (group manager/group member) is encrypted using the EL-Gamal algorithm. Each file always contains a unique file name and is encrypted by its own key which is generated according to the number of bits calculated from each file data. This module provides confidentiality and protects the data from unauthorized access through effective key generation techniques which can secure the system from insider attacks, chosen-plaintext attacks, collision attacks, etc. as well as in data sharing process man-in-the-middle attack.
2. Data file decryption module: Using this module user (group manager/group member) can decrypt and download the files using a decryption key. Here, before downloading the particular data file, take approval request is sent to the data owner. If the data owner accepts the request, the user can decode and download the file by utilizing the key.
3. Rank-Search module: In this module, the user can perform a multi-keyword ranked search over encrypted data files. Users enter query keywords that they are very interested and at last, users get the search result as the topmost relevant document in ranked form as an output.
4. Revocation module: Group Members are revoked by the group manager and admin from the corresponding group using a revocation list when members leave the group or move to a different group. Members are also revoked due to their unauthorized action and after the revocation of any group members, they are not able to access his data.

6 Algorithm

6.1 El-Gamal Algorithm

El-Gamal public key cryptography algorithm is utilized to provide security through effective key generation techniques and encryption strategy. An El-Gamal algorithm is an asymmetric key cryptography technique means it generates two diverse keys for the course of encryption and decryption respectively. An El-Gamal algorithm is an approach to public-key cryptography; it uses a key-based strategy for encrypting data. Public-key cryptography works utilizing calculations that are not difficult to measure one way and hard to measure the opposite way[14]. With respect to, the El-Gamal algorithm will be more difficult in light of the fact that El-Gamal has a convoluted computation to tackle discrete logarithms. The El-Gamal algorithm plays out the encryption cycle on the plaintext blocks which then, at that point, delivers the ciphertext blocks. The level of difficulty in El-Gamal lies in the computation of discrete logarithms on enormous prime modulo. The bigger the number utilized, the harder the discrete logarithms are addressed [15]. The benefit of the El-Gamal algorithm is the generation of keys utilizing discrete logarithms. In the proposed system, the El-Gamal algorithm generates total of four keys in that there is one master key, two public keys, and one private key, due to multiple keys it is quite hard for an unauthorized user to perform malicious activities and get easy access on data. While uploading a file in the cloud, the El-Gamal algorithm is utilized, where each data file is read and according to the number of bits of data file calculate and according to that in that range a random key is generated which is called a master key. Further according to the master key the further public keys and private keys are generated. El-Gamal is a probabilistic, fast, and efficient algorithm [16]. Key generation using the El-Gamal algorithm is given by the following steps:

- Key generation:

Choose a large prime number p such that has a large prime factor and discrete logarithm problem is hard to solve.
Input: Plain text as text data 'M'.
Output: A public key and private key.
Step 1: Initialize the random file from the user as 'M'
Step 2: Select random number 'q' as rime number and alpha as the number of bits of the file data

$$q < 2^{alpha} \qquad (5)$$

Step 3: A large rime number 'p' is generated as

$$p = 2 * q + 1 \qquad (6)$$

Step 4: Select 'g' to be a primitive root of mod p
Step 5: Select randomly 'a' such that, be a member of the group; (0<=a<p-1)
Step 6: Compute

$$b = g^{a} \bmod p \qquad (7)$$

Step 7: Public key = g and b; Master key = p
Step 8: Private key = a
Step 9: Return keys

- Encryption:

First, select the plaintext and resent it as an integer 'm' that wants to encrypt.
Input: Plain text as text data 'M'.
Output: Return the ciphertext.
Step 1: Sender gets the public parameter (g, b, p, m)
Step 2: Select a random integer 'k', k<q, here k is nothing but sender's private key. Then it can generate the secrete key as:

$$K = b^{k} \bmod p \qquad (8)$$

Step 3: Compute

$$C1 = g^{k} \bmod p \qquad (9)$$

$$C2 = Km \bmod p = m * b^{k} \bmod p \qquad (10)$$

Step 4: Return (C1, C2); C1 and C2 are ciphertexts.

- Decryption:

Input: Ciphertext as C1 and C2.
Output: Return the plain text m'.
Step 1: Receiver gets the parameter is (a, p, C1, C2).
Step 2: Compute

$$m' = C2\ (\ C1^{a} \bmod p)^{-1} \bmod p \qquad (11)$$

Step 3: Return m'
Step 4: m' is the plaintext.
Proof to get back plaintext using decryption:

$$m' = C2\ .\ (C1^{a})^{-1} \bmod p \qquad (12)$$

$$m' = m * b^{k}.\ ((g^{k})^{a})\bmod p \qquad (13)$$

$$m' = m * (g^{a})^{k}.\ ((g^{k})^{a})^{-1} \bmod p \qquad (14)$$

$$m' = m \text{ (Original Message)} \qquad (15)$$

6.2 Ranking algorithm

Input: HashMap < double, string >
Output: URL list with weight
Step 1: Read every (k to HashMap)
Step 2: Assess each $Li = \sum_{k=0}^{n}(\text{Hashmap}[k])$
Step 3: Display Li with the greatest weight
Step 4: End for
Step 5: all Li descending order

7 Mathematical Model

S= {a, z, X, Y} Where,
a = The program starts.
X = Program entry.
Input should be keywords in the plain-text file.
Y = Exit the program.
The uploaded file is shared in a group and searching is done using the multi-keyword ranked search method and a proper ranked list of the file is generated according to their similarity score.
z = End of the program.
Extract files from the cloud storage system.
X, Y ∈ U
Let U be the set of systems.
U= {G, GM, GMe, S, D, F}
Where G, GM, GMe, F, S, K, D, F are the elements of the set.
G= Group Name
GM=Group Manager
GMe=Group Member
S=Search using multi-keywords
D=Download file
M=Set of Function/system module which holds the overall system, M= {M0, M1, M2, M3, M4, M5}

- M0: Authentication and Key Generation Process.

El-Gamal security algorithm is executed to generate Encryption and decryption keys and Return public key and private key.

- M1: Upload file with data encryption and data checking.

The file is encrypted using the El-Gamal algorithm using encryption keys. Compute two ciphertexts 'C_1' and 'C_2' using equations (10) and (11).

- M2: Index Generation.

The Index is generated for a file to be uploaded, apply TF-IDF algorithm using the formula given in equation (1), (2) and document vector generate using the formula given in equation (3).

- M3: Sharing:

Data owner (group manager/ group member anybody) who wants to share the data file with the other group members perform sharing operation.

- M4: Ranked Searching

Group members perform a searching operation by sending query keywords of interest. The search result gives the ranked list of a file is generated according to their cosine similarity score between query keywords and stored file and is calculated by using the formula given in equation (4).

- F5: Decryption and downloading

The file in the search list is decrypted using the El-Gamal decryption key compute using equation (12) and downloads files. Return the original file in plaintext.

8 Result and Discussions

The framework is carried out on a java 3-level design structure with INTEL 2.8 GHz i3/i5 processor and 4 GB RAM with public cloud amazon EC2 consol. In the wake of carrying out some pieces of the framework, we got framework execution on an agreeable level. The sharing process of the proposed system is shown by some screenshots of GUI given in below Fig. 3., Fig. 4. , and Fig. 5.

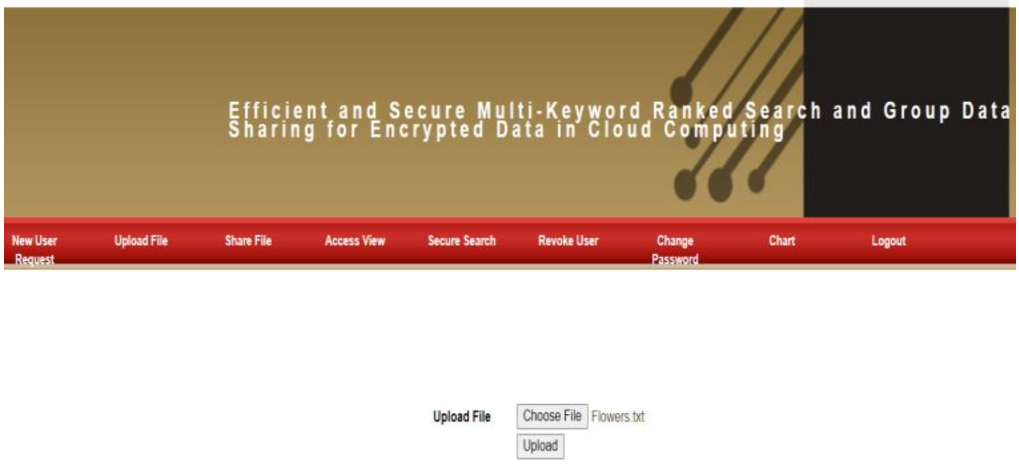
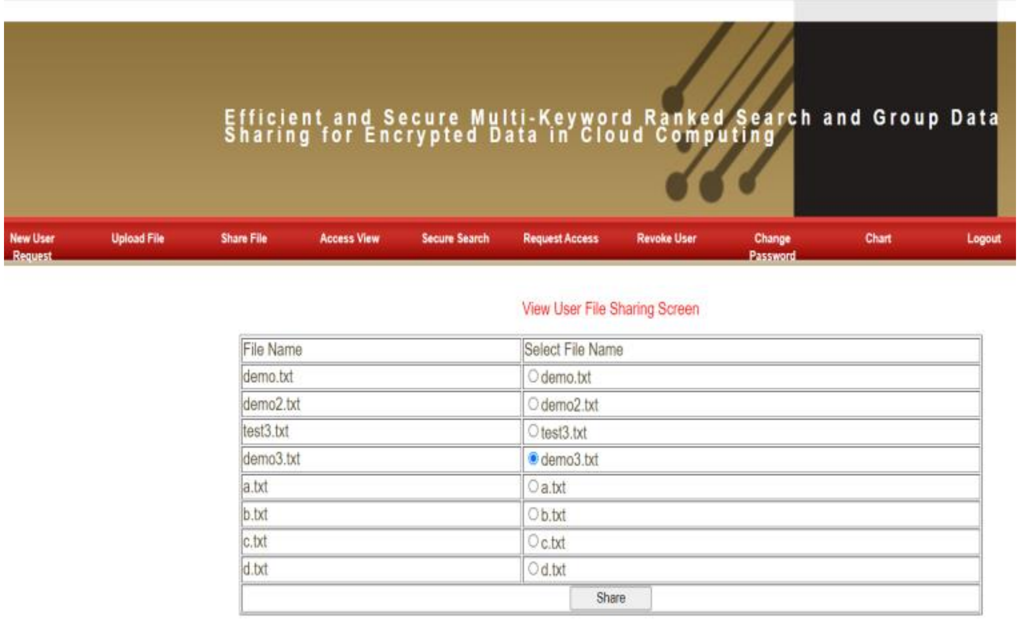


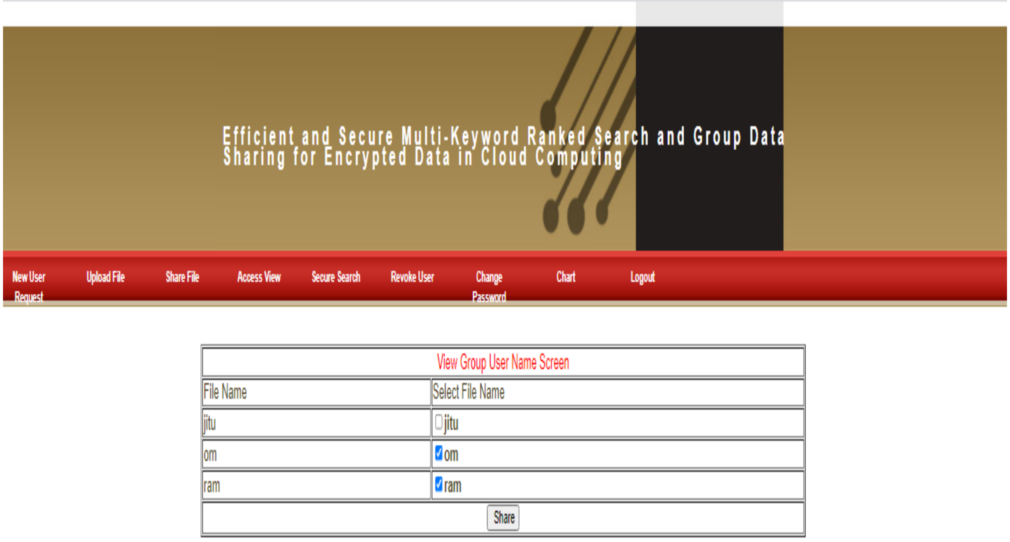
Fig. 6. File upload page



View User File Sharing Screen

File Name	Select File Name
demo.txt	○ demo.txt
demo2.txt	○ demo2.txt
test3.txt	○ test3.txt
demo3.txt	● demo3.txt
a.txt	○ a.txt
b.txt	○ b.txt
c.txt	○ c.txt
d.txt	○ d.txt

Fig. 7. Choose file for sharing by the data owner


Efficient and Secure Multi-Keyword Ranked Search and Group Data Sharing for Encrypted Data in Cloud Computing
New User Request
Upload File
Share File
Access View
Secure Search
Revoke User
Change Password
Chart
Logout

View Group User Name Screen

File Name	Select File Name
jitu	□jitu
om	☑om
ram	☑ram

Share

Fig. 8. File sharing page by the data owner to group member

8.1 Security algorithm evaluation

In our system, we use the El-Gamal algorithm for security purposes. To assess the performance of the proposed security algorithm, consider the speed (time) of an algorithm to encrypt and decrypt the plaintext data files of the different sizes (in KB) required. The obtained result is compared with the ECC model and RSA model. Below Table 1., and Fig. 6. shows comparative analysis.

Table 2. Encryption time performance evaluation

Data Size in KB	Encryption time (Milliseconds)		
	ECC	RSA	Proposed (El-Gamal)
8	52	41	36
16	53	45	40
32	56	53	47
64	67	63	60

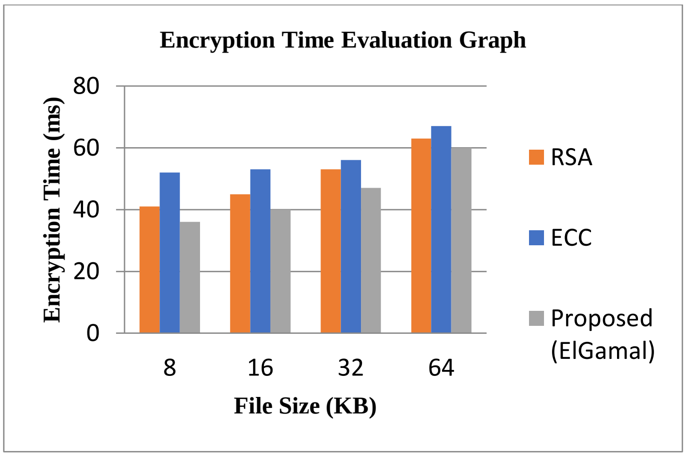


Fig. 6. Encryption Time Evaluation Graph

Encryption time is nothing but the time taken by plain text data files for the encryption process by using the cryptographic algorithm in the cloud framework. The time required for the encryption process generally depends upon the size of the data file and from the above evaluation in Table 3. and Fig. 6. it is clear that as the size of the file will increase the encryption time also increase in each of the algorithms. As compared to ECC and RSA algorithm the proposed system with the El- Gamal algorithm required less time for the encryption process and hence give a better result.

Table 2. Decryption time performance evaluation

Data Size in KB	Decryption time (Milliseconds)		
	ECC	RSA	Proposed (El-Gamal)
8	54	44	40
16	58	49	44
32	64	57	51
64	66	60	57

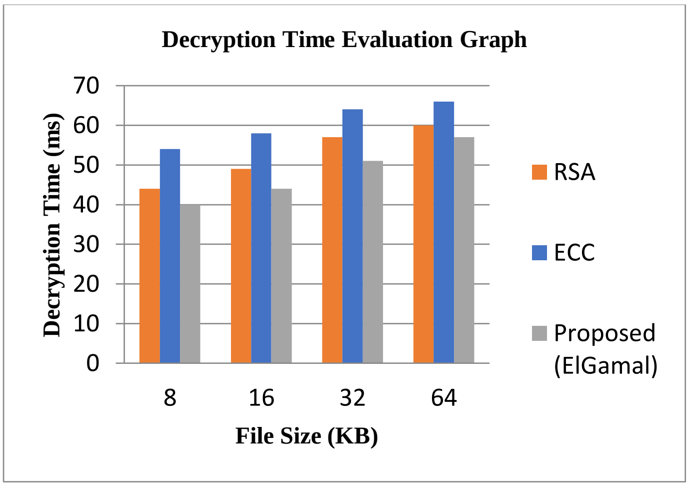


Fig. 7. Decryption Time Evaluation Graph

Decryption time is the time taken by text data files for the decryption process by using the cryptographic algorithm in the cloud framework. The time required for the decryption process generally depends upon the size of the data file and from the above evaluation in Table 2. and Fig. 7., it is clear that as the size of the file will increase the decryption time also increase in each of the algorithms. As the compared to ECC and RSA algorithm the proposed system with the El-Gamal algorithm required less time for the decryption process and hence give a better result.

8.2 Efficiency Evaluation

For the evaluation of the search efficiency of the proposed system, we consider the time required to search keywords of the specific number of groups using the proposed scheme i.e. cosine similarity score method, and compare it with the existing system [7] methodology i.e. evaluate the similarity score using TF-IDF similarity score method. Below Table 3. and Fig. 8. show the comparative evaluation. One of the most important things about the proposed system is it is developed over the real-time data text file uploaded by users, instead of dataset.

Table 3. Search efficiency evaluation

No. of Query Keywords	Search time (Milliseconds)	
	Existing System (TF-IDF)	Proposed System (cosine similarity)
5	36	32
10	45	40
15	76	62
20	122	96

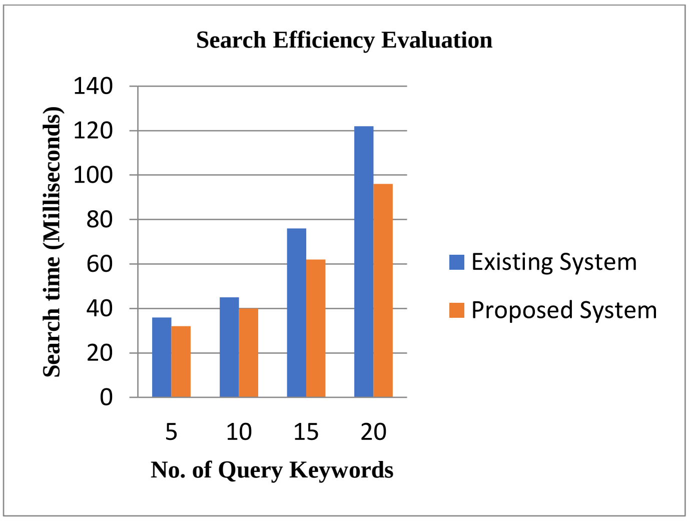


Fig. 8. Search Time Evaluation Graph

Conclusion

Information security and protection are the two significant worries for users while cloud computing. Specifically, applying security worries for multiple owners and furthermore ensuring data protection turns into a difficult assignment. In this paper, a secure group for data exchange and multi-keyword ranked search in a cloud computing scheme is proposed. In our scheme, the data owner could encrypt his private data and share them with a group of data access gadgets at the same time helpfully dependent on the proposed procedure. To create an index and do searches in the encrypted text, several techniques are used. Each data file's index is combined into a single index. This is a secure pursuit convention that enables several data owners to encode files and indexes using separate keys. The cloud server can then combine encoded indexes without knowing any data other current techniques for keyword mapping are less efficient than our index-based search strategy. From the security point of view, we have proposed two secure index schemes with fulfill the security requirement, and also the result analysis shows that the proposed system shows better results. Again efficiency of our system depends on the proposed concept of an index-based search strategy with required less computation time than the linear searching strategy. In the future, we will try to implement an environment system that can focus on multi-keyword ranked search with semantic meaning. Also, we will include operations such as data updating, deletion, and insertion over cloud data under a multi-owner setting.

References

[1] Chandel, Sonali, Geng Yang, and Sumit Chakravarty, "RSA-CP-IDABE: A Secure Framework for Multi-User and Multi-Owner Cloud Environment," Information 11.8 (2020): 382.

[2] Chandel, S.; Yang, G.; Chakravarty, S., "AES–CP–IDABE: A Privacy Protection Framework against a DoS Attack in the Cloud Environment with the Access Control Mechanism," Information 2020, 11, 372.

[3] Y. Miao, X. Liu, K.-K. R. Choo, R. H. Deng, J. Li, H. Li, and J. Ma, "Privacy-preserving attribute-based keyword search in shared multi-owner setting," IEEE Transactions on Dependable and Secure Computing,2019.

[4] Huang, Q.; Yang, Y.; Yue, W.; He, Y., "Secure Data Group Sharing and Conditional Dissemination with Multi-Owner in Cloud Computing," IEEE Transactions Cloud Computing 2019.

[5] Anand, S.; Perumal, V., "EECDH to prevent MITM attack in cloud computing," Digit. Commun. Netw. 2019, vol. 5, pp. 276–287.

[6] Li, Jiayi, et al. "Practical Multi-keyword Ranked Search with Access Control over Encrypted Cloud Data." IEEE Transactions on Cloud Computing (2020).

[7] Xu, Jian, et al. "An Efficient Multi-keyword top-k Search Scheme over Encrypted Cloud Data." 2018 15th International Symposium on Pervasive Systems, Algorithms, and Networks (I-SPAN). IEEE, 2018.

[8] Xue K, Xue Y, Hong J, Li W, Yue H, Wei DS, Hong P. "RAAC: Robust and auditable access control with multiple attribute authorities for public cloud storage." IEEE Transactions on Information Forensics and Security, 2017, vol. 12 no. 4, pp. 953-67.

[9] Kan Yang and Xiaohua Jia, "Expressive, Efficient, and Revocable Data Access Control for Multi-Authority Cloud Storage," IEEE Transactions on parallel and distributed systems, vol. 25, no. 07, July 2014.

[10] Zhongma Zhu and Rui Jiang, "A Secure Anti-Collusion Data Sharing Scheme for Dynamic Groups in the Cloud," IEEE transactions on parallel and distributed systems, vol. 27, no. 1, 2016.

[11] N. Attarpadung, B. Libert, and E. Pana_eu, "Expressive key policy attribute based encryption with constant-size ciphertexts," 2011.

[12] F. Zhang and K. Kim. "ID-Based Blind Signature and Ring Signature from Pairings." In ASIACRYPT 2002, volume 2501 of

Lecture Notes in Computer Science, pages 533547, Springer, 2002.

[13] S. N. Sarode and G. R. Chillarge, "A Review on Secure Techniques for Keyword-Based Search and Data Sharing in Cloud Computing," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 1412-1419, doi: 10.1109/ICICV50876.2021.9388509.

[14] Boomija, M. D., “Secure data sharing through additive similarity based El-Gamal like encryption,” 2016 2nd International Conference on Advances in Electrical, Electronics, Information, Communication, and Bio-Informatics (AEEICB). doi:10.1109/aeeicb.2016.753837

[15] Siahaan, Andysah Putera Utama & Elviwani, Elviwani & Oktaviana, Boni, “Comparative Analysis of RSA and El-Gamal Cryptographic Public-key Algorithms,” 2018, 10.4108/eai.23-4-2018.2277584.

[16] Arockia Panimalar.S, Subhashri.K, “Securing Outsourced Data On Cloud Using El-Gamal Cryptosystem,” International Research Journal of Engineering and Technology (IRJET) Vol. 04, no.10, 2017.

***