Augmenting Privacy For E-Wallet Applications Using Ranked Keyword Search

Shibin David, Jaspher W kathrine

Abstract: Mobile cloud computing is an emerging Internet-based computing system, which enables mobile users to remotely store their data in a server and provide services according to their needs. Many of the devices which use mobile cloud computing are portable these days. Here, Data security and privacy become significant issues in this contest. Before mobile cloud users, a safe data access control mechanism must be given as most customers have to outsource delicate information for storage to the mobile cloud. Nowadays, many of the MNCs and other famous organization uses mobile cloud services for data sharing, data storage, etc. As there are lots of confidential corporate data being shared on mobile cloud servers, it is very important to provide an efficient encryption system to encrypt outsourced data with fine-grained access control. In the mobile cloud, environment encryption can be the best option to secure data out there. The data may be either Private data or Public data. The public data can be shared among trusted clients that provide an open environment to connect with the organization or the data owner. Private data is client's confidential data, and it needs more security and privacy so the data must be transferred in encrypted form. In this paper, various encryption schemes are studies and used in enabling.

Index Terms: mobile cloud computing, data offloading, data outsourcing, data security, privacy, cloud storage, encryption schemes.

1. INTRODUCTION

It is witnessed that there are too many applications which use the mobile cloud environment to store and access mobile cloud data from everywhere over the Internet. Mobile cloud service provides a more advanced facility for the mobile cloud users to store the users' local data in the remote server. Mobile Cloud Service Providers (CSP) is the one which helps the users to access or manage the data stored in the remote data center. Much care is needed during the data processing process, most of the time which will be held in the remote server. Basically, the data which are stored or processed in the mobile cloud computing environment can broadly be classified into two-Private Data and Public Data. Both of these have specific security and privacy requirements, which should be provided by the CSP. The risk rate will be very much high if adequate security measures are not adequately provided for data processing, data storage, and data transmission. One of the main advantages for users and the organizations to use the mobile cloud services is that it is such an easier process to share the data stored in the data server to anyone at any part of the world, provided the authentication capability is provided for that user by the data owner. As technology has shown up a higher growth, there arise challenges which are significant in mobile cloud computing. And, according to each of these difficulties, there must be adequate solutions to prevent all of these difficulties. Security and Privacy of the data are an important factor to be considered. Because anyone can't just say that the CSP which we trust is secure by 100%, always there will be some security issues which needed to be considered as critical. Much more care should be taken as the most important motive of the user to use the mobile cloud service is to share the particular data to anyone at any part of the world. There should be proper authentication capacity for a better mobile cloud service. The process of encryption is a better information-securing solution. Before storing on the wearable cloud, it's better to encrypt the information. The data owner is the one who can give them permission to anyone who wants to access particular data. Data access control should be managed by the data owner. Authentication and better Encryption can provide good security for the data. Data integrity and Data Recovery properties should be shown by a good mobile cloud service in order to give the clients a much more secure environment. Data need to be protected to avoid them from the attacks. As users are able to access the data from anywhere, there are lots of loopholes that could make the attackers' job look easy to access all those private data. So better encryption can give pretty good data privacy and secure environments to share their data across for further processing.

2 Research Background and Literature

2.1 Types of Operations

There are mainly two types of operations of data in the Mobile cloud, including Outsourcing and Offloading. These are the two operations in which the data transmission takes place. And in between these two operations, the data will be processed according to the user requirement. And this operation is called Data processing phase.

2.2 Outsourcing and Offloading

Mobile cloud relies on sharing the computing resources rather than possessing a local copy of it in the servers to handle any applications for an organization or individual [4][5]. This technology does not need much investment for infrastructure and hence serves popular amidst the existing technologies. Many companies are looking for these advantages to be used to the greatest extent. Mobile cloud service makes it possible to access information from anywhere at any time. Mobile cloud computing implements networking of large servers that does processing with some specialized connections. Virtualization [7][8-10] is a technique which enhances the power of mobile cloud computing and supports heterogeneous resources. It allocates resources on demand and allows scaling immediately. Data outsourcing can be defined as the data stored in the mobile cloud storage. But in mobile cloud storage, the resources are stored managed by the third party, which cannot always be trusted. Hence, security and privacy become a concern here. Data outsourcing [6] is a major component for mobile cloud computing because data owners are able to distribute resources to external services for sharing with users and organizations. A crucial problem for owners is how to secure sensitive information accessed by legitimate users only using the trusted services. Mobile cloud computing is emerging as a component of IT outsourcing that enables vendors to offer traditional IT facilities through the mobile cloud on a more flexible basis than may be possible with traditional, fixed infrastructure-based services. Offloading has received a great deal of attention in mobile cloud computing

studies because it has comparable objectives to the evolving mobile cloud computing paradigm, i.e., to surmount mobile devices' shortcomings by augmenting their capabilities with external resources [4-9]. Offloading or enhanced execution relates to a method used to overcome the computing, memory, and battery constraints of mobile phones. Such apps, which can be divided adaptively and components offloaded, are called elastic mobile applications. Basically, this model of elastic mobile apps provides developers with the illusion that they are programming nearly much more energy. In addition, elastic mobile apps can operate as stand-alone mobile apps, but can also adapt internal resources. Which parts of the implementation are executed remotely is chosen at runtime based on the accessibility of resources. In comparison, client/server apps have static partitioning of code, information and business logic between server and client, which is decided in the development

2.3 Types of Encryption

The information may be revealed or altered by any unauthorized access. Special care must be taken to safeguard our delicate information. Secure storage must be achieved in mobile cloud computing. So we adopt cryptographic techniques for secure storage [10]. The information is encrypted by the information proprietor before uploading the information to the mobile cloud. The major feature of cryptographic storage is that the security properties that are described below are accomplished.The data owner applies cryptographic methods to sensitive data to protect the information from unauthorized access. The data owner uploads the encrypted data into the mobile cloud environment. The authorized user may decrypt the data and download the required file.The Strength of Cryptographic Mobile cloud Storage is mostly depending on two factors; they are Confidentiality and Integrity.Confidentiality: Cryptographic Mobile cloud Storage provides Confidentiality as the main characteristics [11-12][16]. The information's were encrypted with the advanced cryptographic techniques, and thus, the secrecy is maintained.Integrity: Mobile cloud Storage provides Integrity to the data, and thus, it prevents any unauthorized people from modifying the data [13][14]. The main components of a cryptographic storage service which can be implemented by using different techniques, out of which, some were designed specifically for mobile cloud storage. At the beginning of the Mobile cloud computing, common encryption Technique like Public Key Encryption was applied. This traditional technique does not provide an expected result as it supports one to one encryption type communication. Public key Encryption isn't extremely scalable. This gave rise to some sophisticated techniques of Encryption [15]. The advanced cryptographic methods include the encryption methods such as searchable Encryption, Symmetric searchable Encryption, Asymmetric Searchable Encryption (ASE) comprises of Homomorphic Encryption, Identity Based Encryption and Attribute-based Encryption, Key policy - Attribute-based Encryption, Ciphertext policy - Attribute-based Encryption, Multi Authority - Attribute-based Encryption and mobile cloud Data encryption standard (DES) Algorithm.

2.3.1 Searchable Encryption

A searchable encryption system [15-19] is implemented at a high point to encrypt the information available in the search index so that it can be concealed from others except for the party providing the approved tokens. A collection of files which consists of full-text index otherwise keyword index considered to generate a search index. The index is encrypted based on a searchable encryption scheme in such a way that,

(i)The pointers to the encrypted documents can be obtained based on the keyword tokens provided.

(ii)If the token is not provided, the index content will be camouflaged. However, the tokens are generated with a full understanding of a hidden key.The retrieval procedure [17] does not reveal the content of the files or the keywords apart from the files that comprise the keyword in common. The previous statement is worth talking about since it is difficult to understand the searchable Encryption that is applicable for security. The studies recognized the file containing the frequent keyword after many searches may have a likelihood of delivering the data to third parties. The server automatically guesses some hypothesis of keywords being searched based on the repeated search from the client search pattern. While searching, some data is leaked, and this data is comparable to the file that the server returns to the client. This information that is leaked and based on leaked information, the server retrieved the file is learned by the provider [18]. (E.g. a file may contain repeated keywords). We can also say that the information leaked to the supplier is based on the service being used, whereas the cryptographic primitives do not disclose it (i.e., Exact keyword matches are used to receive documents). This leakage seems almost vital for both effective and reliable mobile cloud storage service. At worst, the data leaked from the public mobile cloud storage service has much less information. Depending on different scenarios, there exist various types of searchable encryption schemes that can be applied. For example, Symmetric Searchable Encryption (SSE)[30] is implemented for data processing in small enterprise architectures, while Asymmetric Searchable Encryption (ASE) is implemented for large enterprise architecture.

2.3.2 Homomorphic Encryption

Ronald Rivest et al. [19] explain Homomorphic encryption concepts. This scheme is applied in the mobile cloud environment to protect the data. This Homomorphic encryption system enables computations to be executed on encrypted information. It's just the sophisticated cryptographic method. In [20], the major drawback of homomorphic Encryption is explained. It has a slow processing time during the computation.

2.3.3 Identity-based Encryption

Identity-Based Encryption cryptographic scheme has been developed by Shamir [21] in 1984. A significant problem is the inability to create an RSA-based identity-based encryption scheme. Boneh and Franklin created effective identity-based Encryption later in 2001[19].A user identity plays a crucial role in Identity Based Encryption. The sender that sends the signal only requires to understand the identification characteristic of the receiver to deliver the encrypted emails. Email encryption[44][45] is one of the main apps for identity-based Encryption. Key revocation, however, is not attained in Identity Based Encryption.

2.3.4 Attribute-based Encryption

Attribute-based Encryption [22][31-34] is one of the cryptographic techniques used in Mobile Cloud Computing

Environment. Attribute-based Encryption was first placed into use by Sahai and Waters in 2005. The primary focus of this attribute-based encrypted file is to provide safety for the information stored in the mobile cloud.The four steps in Attribute-Based Encryption are Setup, KeyGen, Encrypt, Decrypt [23][26-29]. The KenGen()algorithm is used to generate a user's private key for secret sharing. Authorized end users can decrypt the data using their private key. Attribute-based Encryption comes with access control. In Attribute-Based Encryption, data owner uses a set of attributes to encrypt the data, and only the authorized users who have the predicted, or certain attributes can decrypt the data. This encryption scheme makes the mobile cloud environment better.The different classes of encryption techniques based on attributes are summarized.

2.3.4.1 Key-Policy Attribute-based Encryption

Key Policy-Attribute-Based Encryption [24-25] is introduced by Vipul Goyal and Om Kant Pandey to achieve fine-grained access control in one-to-many communications. The encrypted data is constructed with a set of attributes in Key-Policy Attribute-based Encryption. The entity is entitled to decrypt the Ciphertext if and only if the characteristics produced with the Ciphertext fulfill their personal or secret keys ' access structure [42]. The four steps in Key-Policy Attribute-Based Encryption are Setup, KeyGen, Encrypt, and Decrypt. The algorithms of KeyGen and Decrypt differ from those of the attribute-based encryption. In Key-Policy Attribute-based Encryption, the private key of the user is connected with the access structure [54]. The people may decrypt the information; however, unauthorized access may occur. This can be overcome in the Ciphertext-Policy Attribute-Based Encryption [35] which constructs the access policy in the encrypted data, i.e., Ciphertext and employs a set of attributes to narrate the private key of the user. Also, in some applications that use this scheme, the owner of the data must have a firm belief with the key issuer.

2.3.4.2 Ciphertext Policy Attribute-based Encryption

In 2007 [36], Bethencourt et al. proposed a cryptographic technique named ciphertext policy attribute-based method. The access policy is built with data that has been encrypted. In CP-ABE the Ciphertext is identified with access structure and the private keys with the attributes. In Key-Policy Attribute-Based Encryption, the major disadvantage is that the access policies were not created by the encryptor [52][53]. This [aThis provided a route to the establishment of Ciphertext-Policy Attribute-Based Encryption that enables the encrypted information to be used to build the access policies. The owner who encrypts the data, model the access policy. A suggestion was created to use the CP-AB method. The information proprietor is responsible for identifying access policies. This prevents unauthorized access and promotes safety. Revocation is not effectively achieved in CP-ABE. Thus, modifying the access policies whenever required, is not so simple for the information provider.

2.3.4.3 Multi-Authority Attribute-Based Encryption

The Multi-Authority Attribute-Based Encryption (MA-ABE)[37-38] is also a cryptographic technique consisting of many authorities to manage the attributes and distribute the secret keys. The customer who wants to download the data will request the decryption keys from the power of the Attribute [39-43]. The key generation attribute[51] is one of the algorithms in MA-ABE. This algorithm is run by the organization, and the organization will distribute the keys to the clients. An approved user with the suitable decryption keys can view the data. The algorithms engaged in this system include setup, key generation attributes, central key generation, Encryption, and decryption. This cryptographic scheme handles a number of users. Data confidentiality [39] can be achieved using this type of technique in a mobile cloud environment. Since it is relevant for multiple scenario authorities, this cryptographic technique is most appropriate for apps that involve different sectors. This cryptographic system increases safety and decreases the key management complexity, which is the main benefits.

3 PROPOSED SCHEME

3.1 IDEA BEHIND THE SCHEME

Mobile cloud computing [1][20][23] is an emerging computing model where the data owners are outsourcing their data into the mobile cloud storage. By outsourcing information files to the mobile cloud, it adds many advantages to big businesses as well as individual customers because they can dynamically boost their storage room as and when needed without purchasing any storage systems (Armbrust et al., 2009).

They are:

(1) Users can access the information stored remotely from anywhere at any time and allow approved users to share the information.
(2) Users can be relieved from the local storage management responsibility.
(3) Avoiding outlays on hardware and software costs etc.

Besides all these benefits of outsourced information in the Mobile cloud, there are also some important problems. One of the main problems is the privacy of outsourced data[46-49] in the mobile cloud, i.e., Sensitive information such as e-mail, health records, and public data may leak to unauthorized users (Slocum, 2009; Krebs, 2009) or may even be hacked (Mobile Cloud Security Alliance, 2009). Since the mobile cloud is an open platform, it can be attacked by both malicious insiders and strangers (Haclgiimfi et al., 2002). Mobile cloud service suppliers (CSPs) generally provide data security through mechanisms such as firewalls and virtualization. However, owing to remote mobile cloud storage servers, these mechanisms do not protect the privacy of clients from the CSP itself [50]. A natural approach to preserving the privacy of sensitive data is to encrypt data before outsourcing it and retrieves the data back through a keyword-based search over encrypted data. While encryption protects against illegal access, it considerably improves information owners ' overhead computing, particularly when they have resource constrained mobile devices and big information file sizes.In addition, authorized users want to obtain certain files from the mobile cloud, interact with CSPs, and enable them to function over encrypted data. In order to achieve effective data recovery, it is preferable to receive the necessary files rather than all the files present there.The highlights of the works are represented below.

1. It's a Dynamic Efficient and Secure Privacy-Preserving Approach (Dynamic-ESPPA); It utilizes probabilistic public-key encryption method to decrease owners '

overhead computing during encryption and decryption without leaking any plaintext data.

2. The approach uses ranked keyword search on encrypted data to retrieve the files back. It enables the mobile cloud server to determine whether a given file contains certain keywords and associated relevance score without knowing any information about both the keywords and the files. It significantly decreases overhead communication during the process of file retrieval. It also checks data integrity stored in the mobile cloud.

3.2 System architecture

In this mobile-cloud data storage system, which consisting of three main entities, as illustrated in Fig. 3.

- Data Owner
- Mobile cloud Service Provider (CSP)
- Authorized Users

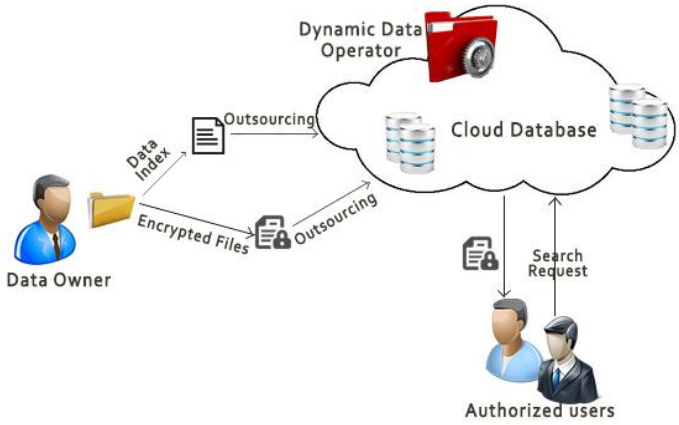

Figure 1 *System architecture*

Data Owner (DO): DO is an entity that has a large amount of data to be stored in the mobile cloud, can be individual user having mobile constrained devices such as smartphones, PDA, TPM chip, etc. Mobile Cloud Service Provider (CSP): CSP is an organization that offers the information owner and users with dynamic data storage facilities and computational resources. Authorized Users (AU): The information patron enables approved customers to use their files and share with the data owner some key material. The approved users would collect the information in an encrypted form from the mobile cloud and obtain the initial information by decrypting it. Following are the typical interactions between these three system entities (Fig. 1).

(1) It is the intention fo the data owner to outsource the data file in encrypted form on the mobile cloud server while retaining the ability to search them through keyword for sufficient data use reasons.
(2) If an approved person wishes to collect the file, send a request for a search to the CSP.
(3) The CSP then searches the files and returns to the user set of file data and hash values.
(4) If any dynamic data operations such as insertion, modification, and deletion for updating documents and corresponding index to be done, there is Dynamic Data Operator, which will make it possible dynamically upon any changes.
(5) The approved user finally checks the integrity and decrypts the documents and receives the respective plaintext.

3.3 System goals

In order to address the privacy of sensitive data stored in the mobile cloud, proposed a dynamic, efficient, and secure privacy-preserving approach with the following goals. Privacy-Preserving: It is to guarantee that there is no way to access sensitive information storage from the mobile cloud for unauthorized parties and malicious insiders. Index Privacy: No data about the respective keywords is leaked by the search index or query index.Efficiency: The above goals should be achieved with less computation and communication overhead. Data Integrity: To detect the modifications or deletions of data and maintain the consistency of data.The information proprietor generates an index for the set of documents in our system and then encrypts both index and files. Later, a query is generated by the approved user and sent to the server. When a request is received by the mobile cloud server, it searches for matching files and gives matched top-k documents to the approved user. The user then decrypts the documents and receives the initial information.

The Dynamic-ESPPA consists of three phases:

1) Setup phase
2) Retrieval Phase
3) Dynamic Data Operation
4) Integrity verification

In the setup phase, the data owner first generates public and private key pairs. Then build the index from various keywords obtained from the set of files, then calculate the score of significance and add to the list of post indexes. The information proprietor then encrypts both to guarantee index and file collection privacy. Finally, the information proprietor distributes to the mobile cloud server the encrypted documents and index. Later, in the retrieval phase, the data owner or authorized user generates trapdoor for a set of keywords and send to the server. Then, the server search for the matched files and their corresponding relevance scores based on the trapdoor. If keywords match with index, the server ranks the matched data-based relevance score and send the data to the user in a ranked ordered manner. Then the data owner or user decrypts the file using private key In integrity verification, and the user verifies the integrity of both file collection and index, i.e., stored data in the mobile cloud is safe or not. The new additional phase is the Dynamic Data Operation phase, where the index and the data get updated dynamically upon any update data operations like insertion, modification, and deletion.

3.4 Working principle of adaptively secure SSE scheme

Secure SSE scheme consists of six functional entities such as,

· Key generation
· Document pre-processing
· Encryption
· Search token generation
· Search
· Decryption

The role of each block is explained below.

a. Key generation: It generates encryption and decryption keys using a simple symmetric key.

b. Document pre-processing: This block provides the necessary feature to pre-process a collection of data and initialize the encryption process. Before encrypting the

document set, the user must pre-process the document set to pull out the keyword and create an index.

c. Encryption: The feature of this block is to encrypt the index and document set. The encryption is carried out using the key generation step encryption keys. It essentially comprises of index encryption and encryption of documents and works as outlined below: Index encryption: This encrypts the keyword collection produced during the pre-processing phase and produces an index/lookup table encrypted. Document encryption: This phase involves encrypting the document present in the document set with a key and keeps it in the database. Search token generation: Search token generation phase is initiated once the user searches for a document that contains a specific keyword. It gives the user/client the ability to generate a search token/trapdoor that can be used on the server to search for encrypted files.

a. Search: The search feature requires the search token and the encrypted index table as input and outputs the searched keyword document list
b. Decryption: After obtaining the encrypted document set containing the searched keyword, the client decrypts the document back.

4 IMPLEMENTATION AND RESULTS

4.1 Implementation method

Based on the framework, the implementation part is divided into four main class structures such as,

- Key Generation
- Encryption
- Search
- Decryption

A. Key generation phase: Two 128-bit AES keys (i.e., keyword encryption key, K1 and document encryption key, K2) are generated and stored. The class of key generation has three techniques.Generate(): two 128-bit AES keys are generated by this technique.Store(): This technique converts the produced keys (encoded) into strings, writes them to a key item and then stores them to a user device place.Read(): This gives the user keys read functionality. The read) (technique requires the main item as input and reads the main values. B. Encryption phase: The encryption is performed at a user device before outsourcing the data to the mobile cloud server. This produces the database that contains the encrypted document set and encrypted index. Before performing encryption, the information must be pre-processed to obtain the keyword and generate an index. The SSE-2 system extracts only a set of separate keywords to produce an index. However, we generate an index of the entire document in our application rather than a finite amount of separate keywords. This produces the database that contains the encrypted document set and encrypted index. Before performing encryption, the information must be pre-processed to obtain the keyword and generate an index. The SSE-2 system extracts only a set of separate keywords to produce an index. However, in our application, we generate an index of the entire document rather than a finite amount of separate keywords.Allocation of document ID: It assigns a unique document ID for each document. The document ID will be allocated, beginning at 0 and will rise sequentially.Keyword extraction: The extraction method of keywords searches for successive sequences of non-blank and non-punctuation characters. Searching for successive sequences of ASCII characters can be readily tweaked. This offers appropriate outcomes for uncompressed English text binary files. It's recommended to convert the file into a text file initially to read the characters from an input document, and we used Apache Tika for this file conversion. The keyword extractor is introduced by pulling characters from the stream, checking if they are appropriate, and accumulating appropriate characters in a keyword. Extra functionality is also given where the number of complete keywords can be limited by choosing a minimum length of characters in the keyword. For an instance, if a user selects the minimum keyword length to be four characters, the keyword extraction algorithm will only remove keywords with 4 or more ASCII characters. The encryption module outputs the encrypted documents after the pre-processing phase. We use AES / CBC / PKCS5 to encrypt the document and index. Encryption class consists of mainly two methods which include,

- encrypt ()
- witeDocIDFileNameMapping ()

Encrypt (): This feature requires all the user-specified route documents as input and encrypts all the documents and also produces an encrypted index. The function retrieves the keywords first for each input file and calculates the 128-bit AES keyword and document ID encryption for each keyword. It then creates an index entry in the encrypted index table to list the encrypted keyword and the corresponding document ID. It uses the set key, value given in the Java Tree Map interface to associate the specified value (document ID) with the specified important (encrypted keyword). Finally, the encrypted index is preserved in the database. Once the indexing is complete, the input files are prepared for encryption. This technique first generates a new file in the database for the encryption of a document and writes the encrypted stream in it. The corresponding document ID is used for each encrypted document as the new filename of the encrypted document.writeDocIDFileNameMapping (): Using this method, a filename mapping object / filename index is created and stored, which stores the identification of an encrypted document and the filename. The hash table is used to map the keys (document ID) to values (filename). This object is stored on the client and used to map the document ID during decryption to get the initial filename and document extension.

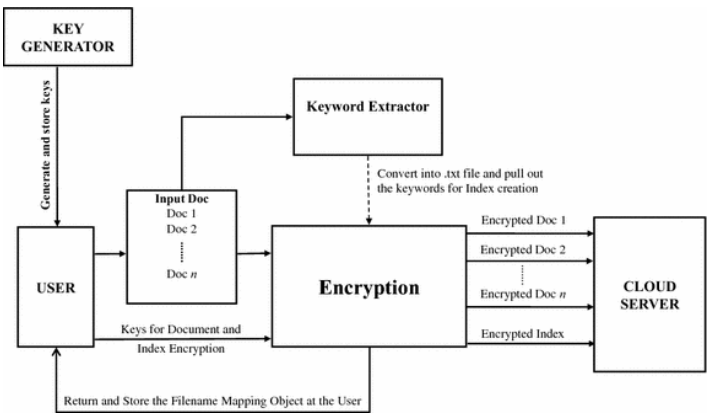

Figure 2 *Encryption module*

A. Search class On the user device, the search function is triggered and carried out on the mobile cloud server. The operation consists of two methods:

- search token ()
- search ()

Search token (): This phase considers the keyword and document ID as input and determines the client device search token. The encryption key for the keyword is used to calculate the encrypted value of the user and returns the created search token. The token of the request will then be sent to the server.Search (): This occurs on the server when the person gets a search application. This technique requires as input the search token and the user database to find out the search result document IDs. It utilizes get() to return the value to which the given key is mapped in the Java Tree Map class, or null if the key is not mapped in this map. Eventually, the server reclaims the appropriate document and sends it back for decryption to the client.

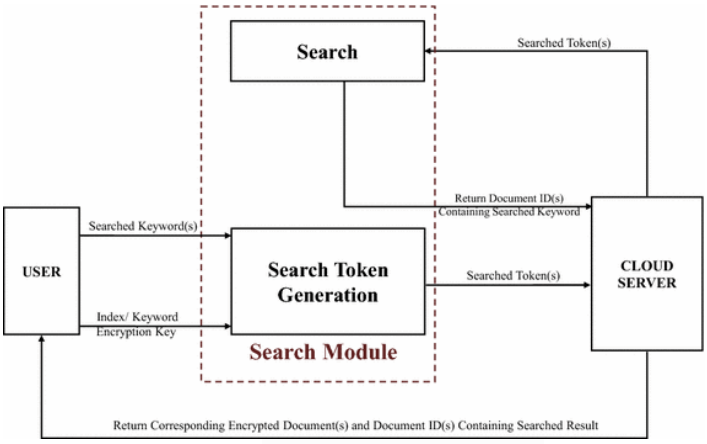

Figure 3 *Search module*

B. Decryption class

This instigates the method of decrypting when the client gets from the server the encrypted file.Decrypt (): Using the user decryption key stored on the client device, this technique decrypts the information in its plaintext form. Once the reader has entered the encrypted document, the program reads the corresponding document ID from the encrypted document filename. It then uses the filename index stored on the user device and uses its ID to get the initial filename and extension for an encrypted document. It then reads the input stream from the encrypted text and uses the respective decryption key to decrypt back the plaintext material. Finally, this technique creates a fresh document with the initial filename and extension that has been obtained and then enters the decrypted plaintext stream into this file and stores it in the user device.

5 PERFORMANCE ANALYSIS

5.1 Performance Metrics

The applications of symmetrical searchable encryption that requires a lot of privacy protection. The performance matrices used to analyze system efficiency are as follows.

5.1.1 Processing Time

Processing includes both Encryptions as well as Decryption. Since our focus is primarily on mobile devices, processing time plays a significant role where it is going to be a significant factor that chooses the need to reduce power consumption. The time for encryption and decryption is very low here in this scheme, which demonstrates the system's efficiency.

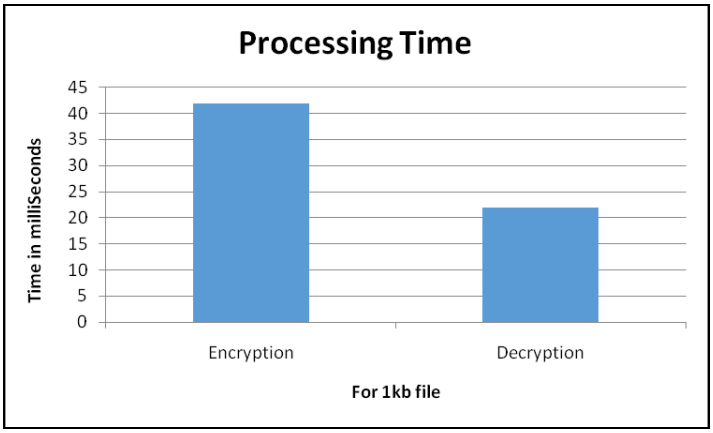

Figure 5 *Processing Time*

5.1.2 Efficiency of Search

The search time includes fetching the posting list in the index, decrypting, and rank-ordering each entry. Our focus is on top-k retrieval. As the encrypted scores are order-preserved, a server can process the top-k retrieval almost as fast as in the plaintext domain. Note that the server does not have to traverse every posting list for each given trapdoor, but instead uses a tree-based data structure to fetch the corresponding list. Therefore, the overall search time cost is almost as efficient as on unencrypted data.

5.1.3 Effect of File Size in processing time

1. Encryption

Encryption is done for Index and file separately. Here we are considering the Encryption of file using Key K2 for our Analysis. And it has been observed that there is no such increase in the Encryption time of files as the file size increases.

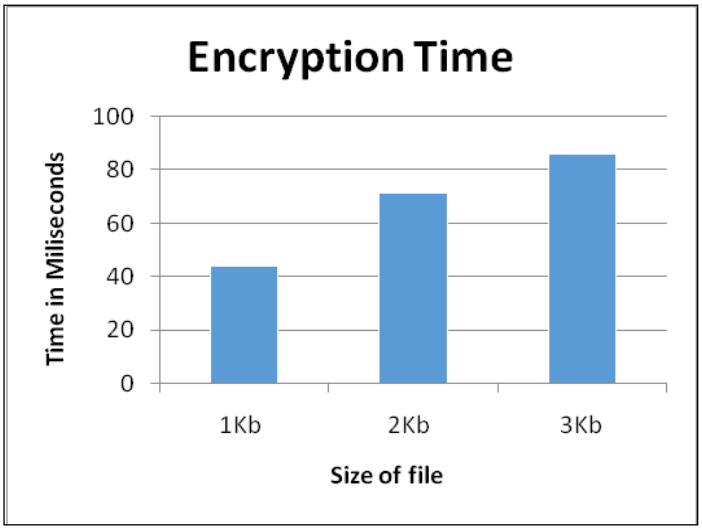

Figure 6 *Encryption Time*

1. Decryption

Processing time for a file is shown in fig 7. As we can analyze here that the decryption time is very less, it will reduce the power consumption by the particular application.

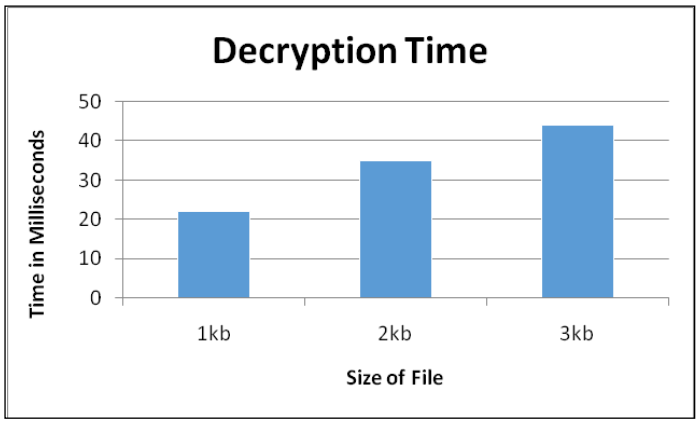

***Figure 7** Decryption Time*

5.1.4 Keyword/Search Token Security

Keywords and Search Token are the words or phrases a DO and Clients will be using for processing index and to retrieve the files through comparing between them. So it is required that both of them should be secure so that even CSP should not be capable of guessing the Keywords or search tokens entered by the user. Here we encrypt them so that there should not be any possibility exists to guess what has been entered. Levenshtein distance is used to measure the similarity between the keyword/search token and encrypted string.

Table 1

Keyword/search token Security

keyword/search token	Encrypted string	Levenshtein distance
Orange	Mp_lec	6
Grapes	Ep_ncq	6
Banana	@_l_l_	6

6 CONCLUSION

Data privacy and information safety are the most significant characteristics to be regarded in portable cloud computing. Encryption can somewhat deals with this problem and find a solution by its own up to a certain limit. But this can give a much high overhead upon computation, if the size of the data is larger, higher the computation overhead will be. Here, Dynamic-ESPPA is being used. It can reduce the overhead of the users by its dynamic property. In the environment, there may be lots of data updating should be done. If the user by itself is trying to do the updating, the overhead will be much higher. So, this method gives owner/authorized user the provision to do it dynamically right in the mobile cloud itself rather than again decrypt the file, then update and then outsource again. So, this method seems to be much better which can reduce the user overhead.

REFERENCES

[1] Sahai, Amit, and Brent Waters. "Fuzzy Identity-Based Encryption." N.p., 2010. 457–473.

[2] Girish, and H D Phaneendra. "Identity-Based Cryptography and Comparison with Traditional Public Key Encryption : A Survey" International Journal of Computer Science and Information Technologies 5.4 (2014): 5521–5525.

[3] Wang, Cong, Qian Wang, and Kui Ren. "Towards Secure and Effective Utilization over Encrypted Cloud Data." Proceedings of International Conference on Distributed Computing Systems. N.p., 2011. 282–286.

[4] Bethencourt, John, Amit Sahai, and Brent Waters. "Ciphertext-Policy Attribute-Based Encryption." Proceedings of IEEE Symposium on Security and Privacy. N.p., 2007. 321–334.

[5] Chase, Melissa, and Melissa Chase. "Multi-Authority Attribute Based Encryption." Theory of Cryptography, 4th Theory of Cryptography Conference 4392 (2007): 515–534.

[6] Chase, Melissa, and Sherman S.M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption." Association for Computing Machinery (ACM), 2009. 121.

[7] Lewko, Allison, and Brent Waters. "Decentralizing Attribute-Based Encryption." Lecture Notes in Computer Science. 6632 LNCS. N.p., 2011. 568–588.

[8] Li, Jin et al. "Securely Outsourcing Attribute-Based Encryption with Checkability." IEEE Transactions on Parallel and Distributed Systems 25.8 (2014): 2201–2210.

[9] Zhang, Yinghui et al. "Ensuring Attribute Privacy Protection and Fast Decryption for Outsourced Data Security in Mobile Cloud Computing." Information Sciences 379 (2017): 42–61.

[10] Li, Hongwei, Yuanshun Dai, and Bo Yang. "Identity-Based Cryptography for Cloud Security." eprintiacrorg (2011): 1–9.

[11] Ostrovsky, Rafail, Amit Sahai, and Brent Waters. "Attribute-Based Encryption with Non-Monotonic Access Structures." Proceedings of the 14th ACM Conference on Computer and Communications Security - CCS '07. New York, New York, USA: ACM Press, 2007. 195.

[12] Wang, Shulan et al. "An Efficient File Hierarchy Attribute-Based Encryption Scheme in Cloud Computing." IEEE Transactions on Information Forensics and Security 11.6 (2016): 1265–1277.

[13] Wang, Guojun et al. "Hierarchical Attribute-Based Encryption and Scalable User Revocation for Sharing Data in Cloud Servers." Computers and Security 30.5 (2011): 320–331.

[14]] Xhafa, Fatos et al. "Privacy-Aware Attribute-Based PHR Sharing with User Accountability in Cloud Computing." Journal of Supercomputing 71.5 (2015): 1607–1619.

[15] Zhou, Lan, Vijay Varadharajan, and Michael Hitchens. "Achieving Secure Role-Based Access Control on Encrypted Data in Cloud Storage." IEEE Transactions on Information Forensics and Security 8.12 (2013): 1947–1960.

[16] Tysowski, Piotr K., and M. Anwarul Hasan. "Hybrid Attribute-and Re-Encryption-Based Key Management for Secure and Scalable Mobile Applications in Clouds." IEEE Transactions on Cloud Computing 1.2 (2013): 172–186.

[17] Huang, Dijiang et al. "MobiCloud: Building Secure Cloud Framework for Mobile Computing and Communication." Proceedings - 5th IEEE International Symposium on Service-Oriented System Engineering, SOSE 2010. N.p., 2010. 27–34.

[18] A. Balu and K. Kuppusamy. 2014. An expressive and provably secure Ciphertext-Policy Attribute-Based Encryption. Inf. Sci. 276, C (August 2014), 354-362. DOI=http://dx.doi.org/10.1016/j.ins.2013.12.027

[19] Attrapadung, N., Libert, B.: Functional encryption for inner product: Achieving constant-size ciphertexts with adaptive security or support for negation. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 384–402. Springer, Heidelberg (2010)

[20]]H. Cheng, C. Rong, Z. Tan, Q. Zeng, "Identity based encryption and biometric authentication scheme for secure data access in cloud computing", Chinese Journal of Electronics, Vol.21,No.2, pp.154–259, 2012.

[21]]Multiple Biometric Security in Cloud Computing; Pugazhenthi, B.Sree Vidya,; https://www.ijarcsse.com/docs/papers/Volume_3/4_April2013/V3I4-0361.pdf

[22] Bastia, Abhijit et al. "Service Composition Using Efficient Multi-Agents in Cloud Computing Environment." Advances in Intelligent Systems and Computing. 308 AISC. Springer Verlag, 2015. 357–370.

[23] P. Tiwari and A. Saklani, "Role of biometric cryptography in cloud computing", International Journal of Computer Applications, vol. 70, pp. 34-38, 2013.

[24] Multivariate Authentication and Encryption Scheme for Data Privacy in IoT Healthcare Monitoring Er. Kritika, Dr. Harjit Pal Singh, Er. Narinder Pal Singh, Er. Mamta www.onlinejournal.in/IJIRV2I8/091.pdf

[25] H. Deng , Q. Wu , B. Qin , J. Domingo-Ferrer , L. Zhang , J. Liu , W. Shi , Ciphertext-policy hierarchical attribute-based encryption with short ciphertexts, Inf. Sci. 275 (2014) 370–384.

[26] Tysowski PK, Hasan MA (2011) Re-encryption-based key management towards secure and scalable mobile applications in clouds. In: IACR cryptology eprint archive, p 668.

[27] Vipul Goyal, Abhishek Jain, Omkant Pandey, and Amit Sahai. 2008. Bounded Ciphertext Policy Attribute Based Encryption. In Proceedings of the 35th international colloquium on Automata, Languages and Programming, Part II (ICALP '08), Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz (Eds.). Springer-Verlag, Berlin, Heidelberg, 579-591.

[28] Brent Waters. 2011. Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. In Proceedings of the 14th international conference on Practice and theory in public key cryptography conference on Public key cryptography (PKC'11), Dario Catalano, Nelly Fazio, Rosario Gennaro, and Antonio Nicolosi (Eds.). Springer-Verlag, Berlin, Heidelberg, 53-70.

[29] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. 2006. Attribute-based encryption for fine-grained access control of encrypted data. Proceedings of the 13th ACM conference on Computer and communications security (CCS '06). ACM, New York, NY, USA, 89-98.

[30] Ling Cheung and Calvin Newport. 2007. Provably secure ciphertext policy ABE. Proceedings of the 14th ACM conference on Computer and communications security (CCS '07). ACM, New York, NY, USA, 456-465.

[31] Kao Zhao, Hai Jin, Deqing Zou, Gang Chen, and Weiqi Dai. 2013. Feasibility of Deploying Biometric Encryption in Mobile Cloud Computing. Proceedings of the 2013 8th ChinaGrid Annual Conference (CHINAGRID '13). IEEE Computer Society, Washington, DC, USA, 28-33.

[32] Liang, Hongbin et al. "Resource Allocation for Security Services in Mobile Cloud Computing." 2011 IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS 2011. N.p., 2011. 191–195.

[33] Yadav, Dipali S, and Doke Kanchan. "Mobile Cloud Computing Issues and Solution Framework." International Research Journal of Engineering and Technology 11 (2016): 2395–56.

[34] Zhang, Yuan et al. "To Offload or Not to Offload: An Efficient Code Partition Algorithm for Mobile Cloud Computing." 2012 1st IEEE International Conference on Cloud Networking, CLOUDNET 2012 - Proceedings. N.p., 2012. 80–86.

[35] Zhang, Lan et al. "POP: Privacy-Preserving Outsourced Photo Sharing and Searching for Mobile Devices." Proceedings - International Conference on Distributed Computing Systems. Vol. 2015-July. Institute of Electrical and Electronics Engineers Inc., 2015. 308–317.

[36] Singh, Aarti & Malhotra, Manisha (Associate & Bm, Mmict &, Mullana). "Agent Based Framework for Scalability in Cloud Computing." International Journal of Computer Science & Engineering Technology (IJCSET) Agent 3.4 (2012): 41–45.

[37] Huang, Dijiang et al. "Secure Data Processing Framework for Mobile Cloud Computing." 2011 IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS 2011. N.p., 2011. 614–618.

[38] Imorsy, Mohemed, John Grundy, and Amani S. Ibrahim. "Collaboration-Based Cloud Computing Security Management Framework." Proceedings - 2011 IEEE 4th International Conference on Cloud Computing, CLOUD 2011. N.p., 2011. 364–371.

[39] Ristenpart, Thomas et al. "Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds". Proceedings of the 16th ACM conference on Computer and communications security (2009): 199–212.

[40]] Wang, Cong et al. "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing." Proceedings - IEEE INFOCOM. N.p., 2010.

[41]] Tang, Jun et al. "Ensuring Security and Privacy Preservation for Cloud Data Services." ACM Computing Surveys 49.1 (2016): 1–39.

[42]] Green, Matthew, Susan Hohenberger, and Brent Waters. "Outsourcing the Decryption of ABE Ciphertexts." 20th USENIX conference on Security (2011): 34–34

[43]] Phaphoom, Nattakarn et al. "A Survey Study on Major Technical Barriers Affecting the Decision to Adopt Cloud Services." Journal of Systems and Software. Vol. 103. Elsevier Inc., 2015. 167–181.

[44]] Khan, Afnan Ullah et al. "Security Risks and Their Management in Cloud Computing." CloudCom 2012 - Proceedings: 2012 4th IEEE International Conference on Cloud Computing Technology and Science. N.p., 2012. 121–128

[45] Julisch, Klaus, and Michael Hall. "Security and Control in the Cloud." Information Security Journal 19.6 (2010): 299–309.

[46] Yusop, Zulkefli Mohd, and Jemal H. Abawajy. "Analysis of Insiders Attack Mitigation Strategies." Procedia - Social and Behavioral Sciences 129 (2014): 611–618.