Journal of Excellence in Computer Science and Engineering, Vol. 1(2) 2015, pp. 11-22

RESEARCH ARTICLE

A Secure Steganographic Method for Efficient Data Sharing in Public Clouds

[*]V.G Gopika[1], Neetha Alex[2]

[1]PG Student, Department of Computer Science and Engineering, TKM Institute of Technology, Kollam, Kerala, India.

[2]Professor, Department of Computer Science and Engineering, TKM Institute of Technology, Kollam, Kerala, India.

Received-12 November 2015, Revised-13 December 2015, Accepted-15 December 2015, Published-29 December 2015

ABSTRACT

Data relaying process through internet has become more faster which makes it easier to send the data precisely to the destination. Security and transmission time are the most important factors of data transmission. The proposed system improves these factors using the concepts of cryptography and steganography. In this paper, the data hiding technique is implemented in encryption-decryption domain, by joining the concepts of steganography and public key cryptography using the sequence encryption-compression then decompression-decryption. The proposed system uses the mediated certificate less encryption (mCL-PKE) scheme which is combined with the data hiding scheme for providing more confidentiality in public key cryptography. The main idea is to implement the separable reversible data hiding scheme into the mCL-PKE scheme where the cover media decryption and hidden data freeing are separated based on the availability of keys. Among the various steganographic algorithms like Least Significant Bit (LSB) algorithm, Jsteg and F5 algorithms, the proposed system uses F5 algorithm which possess the matrix encoding technique that provides high steganographic capacity and can also prevent visual and statistical attacks.

Keywords: Cloud computing, Cryptography, Separable reversible data hiding, Data security, Confidentiality.

1. INTRODUCTION

Organizations have been adopting public cloud services like Dropbox and Microsoft Skydrive for managing their data due to the greater benefits of public cloud storage. For the widespread support of cloud storage, the public cloud storage model must solve the risky issue of data confidentiality. Steganography is the practice of masking a file, message, image, or video within another file, message, image, or video. When it is mandatory to send the confidential data over an insecure and bandwidth-constrained channel it is traditional to encrypt as well as compress the cover data and then embed the mandatory data into the cover data. The advantage of steganography over cryptography is that the purposeful secret message does not attract attention to itself as an object of enquiry.

Cryptography is the action of protecting the contents of a message alone while steganography is concerned with masking the fact that a secret message is on transmission, as well as concealing the contents of the message. In this paper cryptography is combined with steganography for providing better security and reducing the transmission time. This works on the sequence encryption-compression then decompression-decryption for secure data sharing in public clouds.

The existing systems work on the basis of the mediated certificateless encryption (mCL-PKE) scheme which is inefficient due to the use of expensive pairing operations. Also they are more vulnerable against the partial

*Corresponding author. Tel.: +919020792998
Email address: mail4gopika19@gmail.com (V.G.Gopika)
Double blind peer review under responsibility of DJ Publications
http://dx.doi.org/10.18831/djcse.in/2015021002

decryption attacks. So to resolve these performance and security issue problems an enhanced mCL-PKE has been used without using the pairing operations which resolves the key escrow problem in ID based encryption and certificate abrogation problem in public key cryptography. This scheme is similar to Proxy Re-Encryption (PRE) where the data encrypted using the data owner's public key is allowed to be decrypted by using several private keys which acts like a proxy at the receiver end. But our scheme doesn't perform any type of transformations on cloud before the decryption of public keys. Here the cloud simply acts as storage for these proxy keys. In this mediated scheme the administrator encrypts the sensible information using the cloud generated user's public keys based on the access control policies and uploads the encrypted information to the cloud. The cloud partially decrypts the encrypted data for the users after successful authorization. Using the user's private keys, they can subsequently decrypt the partially decrypted data completely. But the Re-Encryption scheme faces some problems during the retrieval of data. They do not retrieve the complete keyword related full document. In a selective manner the documents are retrieved. Thus to resolve this problem the Re-Encryption scheme has been replaced with reversible data hiding techniques.

The reversible data hiding techniques has been classified as separable reversible data hiding and reversible data hiding (non-separable) techniques on the basis of key distribution. In non-separable reversible data hiding technique at first the administrator encrypts the original uncompressed image with an encryption key for producing an encrypted image. Then the data hider embeds additional data into the encrypted image using a data hiding key. The receiver in turn decrypts the encrypted image using the encryption key and can further extract the additional embedded data. The major disadvantage with this technique is that it is compulsory to have both the encryption and data hiding keys for retrieving the data. The proposed work uses the separable reversible data hiding technique which resolves the necessity of having both the keys. The principal notion of this technique is that it consists of three procedures. Firstly to encrypt the cover media, secondly to hide the data and thirdly to get the data as well as cover media as per provisions. Here the administrator encrypts the original uncompressed image using an encryption key which produces an encrypted image. By using a data hiding key the data hider compresses the least significant bit of the encrypted image and create some space to accommodate the additional data. At the receiver end according to the data hiding key, the data embedded can be easily retrieved from the encrypted image which also contains the additional data. Since the data embedding only affects the least significant bits of the encrypted image, an image similar to the original one can be retrieved by decrypting them with the encryption keys. The receiver side has three cases. If the receiver has only the data hiding key, he can extract the embedded or hidden data from the encrypted image containing additional data. Case two is if the receiver has only the encryption key, he can retrieve the original image without removing or freeing the data embedded in the encrypted image. Case 3 is if the receiver has both the keys, he can retrieve the data hidden as well as the original image from the encrypted image.

2. BACKGROUND AND RELATED WORK

To support a fine grained encryption based on access control, an approach which encrypts the varying set of data items for which the same access control policies are applied with different symmetric keys and provide users with either the relevant keys [1] or the capability to attain the keys [2] is used. In order to generate the key approaches that reduce the number of keys to be managed is recommended. In general the symmetric key based mechanism is cost effective for key management. Thus, to reduce the overhead of key management an alternative is to use the public key cryptosystem. To overcome these problems an Identity-Based Public Key Cryptosystem (IB-PKC) was introduced. But it suffers from the key escrow problem as the user's own private key is revealed to a third party. An Attribute Based Encryption (ABE) has been proposed [3] which encrypts each data item based on their access control policies applicable to the data. But, in addition to the key escrow problem ABE faces the revocation problem as the private keys provided to the existing users should be updated whenever a user is revoked. An approach based on ABE

utilizes the PRE (Proxy Re-Encryption) to handle the abrogation problem of ABE even though the defined approach is based on the pairing based cryptography. So to resolve these problems a new cryptosystem named Certificateless Public key Cryptography (CL-PKC) has been proposed [4]. Based on this scheme, the Certificateless Proxy Re-Encryption (CL-PRE) scheme has been proposed for secure data sharing in public cloud environments. It has been introduced to resolve the key escrow problem and certificate revocation problem and this scheme relies on pairing operations which are computationally expensive [5].

Security Mediated Certificateless (SMC) cryptography [6] has been proposed, which allows more insubstantial versions of mediated cryptography while maintaining the ability for instantaneous abrogation of keys. Reference [7] describes a strongly secure CL-PKE without pairing operations. Previously proposed CL-PKE schemes could not resolve the key abrogation problem. A mediated CL-PKE without pairings has been proposed [8], but it is found to be insecure against the partial decryption attacks because in requesting the partial decryption there security model doesn't consider the conditions of the adversary.

To address the shortcomings of such previous approaches, a mediated Certificateless Public Key Encryption (mCL-PKE) scheme [9] which does not utilize pairing operations has been proposed. This scheme reduces the computational overhead by using a pairing free approach. The computational cost for decryption at the users end is reduced and can efficiently manage keys and user revocations. In this approach, each user only needs to maintain his or her public or private key pairs and also the private keys of users does not require any change.

Mostly the reversible data hiding techniques are not separable and they are all not based on encryption-decryption domain. These data hiding (steganography) techniques mainly include the following actions like compression-decompression, encryption-decryption, data insertion and data freeing which provides greater confidentiality and authenticity using the key generation center and so on. The conventional way of transmitting excessive data is to compress the data for reducing the redundancy problem and then encrypt the compressed data. At the receiver end, to recover the original cover data, the decryption and decompression operations are performed in a well-organized manner. But for some of the applications, they require a highly confidential transmission. For this, the network operators who are responsible for the channel resource allocation for the data transmission should keep the information confidential which means that the sender should encrypt the original data. At the receiver end, a decoder harmonizes decompression and use decryption functions.

There are several approaches for the compression and decompression of the encrypted data that has been developed. When it is desired to transmit excessive data over an insecure and bandwidth constrained channel, a normal routine is to first compress the data and then encrypt it. But most of the existing data hiding techniques are not reversible and they are carried out in spatial domains. Hence for data hiding, a technique named reversible data hiding [10] is used in which we can extract data precisely and after which the original cover content can be perfectly recovered. This data hiding technique has certain advantages like distortion free and lossless or invertible. Here the reversible data hiding algorithm utilizes the zero or least points of histogram of an image which is referred to as the histogram shift mechanism. The algorithm works by slightly modifying the pixel gray scale values for data insertion into the image and has greater data embedding capacity. It has been successfully applied to a wide range of images like medical imaging, texture imaging and also on the images of the Corel Draw database. The main objective of this paper is to work on the concept in which text is used as the hidden data. Also no plain spatial domain is used. This makes a huge amount of data to be hidden within a cover media and the quality is evaluated using different interpretations. The three main procedures of separable reversible data hiding is to first encrypt the cover media, secondly to hide the data and finally to get the data as well as the cover media as per the provisions. The lossless generalized–LSB data embedding method make use of the redundancy in a cover by performing a lossless compression which creates a sparse space for data embedding [11].

Reference [12] presents a similar reversible (lossless) data hiding technique named reversible data hiding in encrypted

image which permits the exact recovery of the original signal together with the freeing of the embedded information. After encrypting the entire data of an uncompressed image using a stream cipher, for the additional data insertion or embedding usually the well-known LSB (least significant bit) method is used. This slightly modifies a small portion of the encrypted data. Using the encryption key, the encrypted image containing additional data is decrypted. With the data hiding key, the embedded data can be successfully retrieved with the aid of spatial correlation in the natural image and the original image can be perfectly recovered. Even though someone comes to know about the encryption key, he can decrypt the original image and can detect the presence of hidden data using LSB steganalytic method. But if he doesn't come to know about the data hiding key, it is impossible to extract the additional data and also to recover the original image. In addition to the exact recovery of the original signal they are mainly used for the authentication of data like images, video etc. The main application of this data hiding technique is in IPR (Intellectual Property Rights) protection and authentication. This technique takes advantage of low computation complexity.

A separable reversible data hiding in encrypted image [13] scheme uses image as a cover medium. In this scheme the sender or the content owner or the administrator encrypts the cover image using an encryption key. Then using a data hiding key, the data hider compresses the least significant bits of the encrypted image. This is done to create enough space to accommodate some secure data. At the receiver end there are three cases. Case one is that if the receiver is having only the data hiding key, he can extract the embedded or the hidden data from the encrypted image containing extra data. Case two is if the receiver has only the encryption key, he can retrieve the original image without freeing the data embedded in the encrypted image. Case three is if the receiver has both the keys, he can retrieve the data hidden as well as the original image from the encrypted image. Hence the two actions such as the exact recovery of the hidden data and recovery of cover media are separated. As it is both reversible and separable in nature, this technique is named as "Separable Reversible Data Hiding". In reference [14] the steps are reversed by first encrypting and then compressing without affecting either the compression efficiency or the security.

Reference [15] discusses the problem of transmitting unnecessary data over insecure, bandwidth constrained communication channels. Here for data hiding, Rc4 algorithm has been proposed to implement separable reversible data hiding in the encrypted image. Rc4 is a stream cipher and also a symmetric key algorithm. For both encryption and decryption same algorithm is used as the data stream is simply XORed with the generated key pattern. Inaddition, the key stream is completely independent of the plain text used. A variable length key from 1 to 256 bit has been used to initialize a 256 bit state table. The purpose of state table is for the succeeding generation of pseudo-random bits and to generate a pseudo random series which is XORed with the plaintext to get the ciphertext. Using the encryption key, the random bits are generated for the pseudo random sequence. Comparing with other algorithms, this scheme shows a greater accuracy in the recovery of original image.

[17] used privacy preserving protocols with two levels of privacy. Security for shared data in the case of multiple users is also discussed. As the data is embedded in normal form i.e. without encrypting, data in previous systems [10, 13, 15, 20] data has less security as compared to image. Since the data has more priority than image, to overcome these problems a new separable reversible encrypted data in encrypted image using AES algorithm [16, 18] and lossy compression technique [19] has been used. In this scheme the sender encrypts image and data separately using the AES algorithm and hides the encrypted data in encrypted image using the Least Significant Bit (LSB) technique. Here the system auto generates the 3 respective keys. The sender can transmit the file through the existing mail system and the receiver can perform the operation according to the availability of keys [20]. In [21] the major objective is to describe the concept of separable reversible data hiding using a new approach which describes about improving the performance after increasing the size of payload. Through this approach it is possible to hide large amount of data without affecting either the compression efficiency or security.

3. SYSTEM OVERVIEW

In this section a detailed description of the proposed system for preserving privacy using the mCL-PKE scheme as well as the separable reversible data hiding scheme has been described. The main purpose is to provide or construct a practical solution to the problem of sharing sensitive data in public clouds. The cloud mainly acts as a secure storage. In addition to it, it acts as a key generating center.

The new approach of the proposed system consists of four entities viz steganography, data owner, cloud and users. The data owner attains the sensitive information which it wants to share with the authorized users. This is done by storing the sensitive information in the public cloud which is assigned with the storage purpose and requests the cloud to partially decrypt the encrypted content when the user requests the data. The cloud consists of three main services such as an encrypted content storage, a Key Generating Center (KC) and a Security Mediator Server (SMS). The Key Generating Center (KC) generates the public or private key pairs for each user. Accordingly the Security Mediator Server (SMS) acts as a security mediator for each data request and partially decrypts the encrypted data for the authorized users. The cloud trustily performs the key generation as well as the security mediator services correctly and also provides trusted confidentiality of the content and the key abrogation. This approach allows one to have most of the key generation and the management functionality deployed in untrusted cloud. The mCL-PKE scheme used here does not have the problem of key abrogation and thus it is impossible to identify the private keys of users. Figure A1 illustrate the system overview.

The proposed scheme consists of mainly five phases such as 1)Setup Cloud, 2)User Registration, 3)Data Encryption with Steganography, 4)Data Uploading and 5)Data Retrieval and Decryption. The detailed description of these phases was given in the preceding sections.

4. WORKING OF PROPOSED SYSTEM

4.1. Setup cloud

The setup operation is performed by the Key Generating Center (KC) in the cloud similar to the basic algorithm [9], which takes a security parameter as input and returns the system parameters which are publicly available to all users and a secret master key. These operations are an one-time task.

4.2. User registration

The new users must register for the system and in turn the administrator will provide authorization for the registered users. Firstly the users must login to perform operations and the records of the login session for each user will be stored in the administrator. The records will contain the user id, name, phone and mail id. Internally for this each user generates the public and private key pair. They are generated using the SetPublicKey and SetPrivateKey operations [9]. The algorithm runs for each user. Firstly the private key generation is done. Next the public key generation is done. After generating the public and private key pair, the user sends public keys and identity to the key generating center in the cloud. The KC in turn generates a public key and two partial keys for the user. The partial keys generated are used for the partial decryption of encrypted data. One of the partial key referred as SMS-key is stored at the SMS in the cloud and the other partial key referred as U_k key is provided to the user for the complete decryption of data. The public key referred as KC-key consists of the KC generated public key as well as the user generated public key which is used to encrypt the data.

4.3. Data encryption with steganography

The public key referred as KC-key is provided to the data owner from the KC in the cloud. The data owner then performs the hiding process of data items and symmetrically encrypts these data items for which the same access control policies are applied. This is done using a random symmetric key. The data owner encrypts each data item only once and then mCL-PKE scheme encrypts the random symmetric key using its public key. In addition the data owner downloads the public keys of users for generating the mediator keys. The proposed scheme combines the separable reversible data hiding scheme [13] and cryptography for providing better security. Here the data owner encrypts the cover image using an encryption key. Then using a data masking key, the data masker compresses the least significant bits of the encrypted image

using the F5 algorithm. This algorithm uses matrix encoding which takes greater advantage in the reduction of number of changes needed to embed a message of certain length. In addition to this it avoids the chi-square attack as it doesn't replace or exchange the bits. The resistance is higher for both statistical and visual attacks and has greater embedding capacity as compared to the LSB and JSTEG algorithms. This is done to create enough space to accommodate the data securely.

4.4. Data uploading

After the data hiding and encryption operations, the data along with the access control list and the owner generated mediator keys are uploaded to the cloud for storage purpose. The encrypted content is stored in the secured storage center in the cloud and the access control list which is approved by the data owner and the mediator keys are stored in the Security Mediator Server (SMS) in the cloud.

4.5. Data retrieval and decryption

When a user wants to get some data, firstly the user sends a request to the SMS for obtaining the partially decrypted data. In turn the SMS checks whether the user is in the access control list or not. If the user's KC-key encrypted content is available in the cloud storage center i.e if the verification is successful, the SMS retrieves the encrypted content from the storage center in the cloud and then partially decrypts the retrieved data using the SMS-key for the user. This partial decryption takes advantage of load reduction on users. The user then fully decrypts the data using its private key, partial private key U_K and mediator key.

5. RESULTS AND DISCUSSIONS

This section defines the experimental results of the proposed system. Some discussions based on the graphs are also presented here. The experiments were performed on a machine running 64 bit with an Intel®Core™i3-3217U CPU @ 1.80GHz and 4.00GBytes memory. The proposed system is implemented using J2SE with MySQL as database. According to the experimental results, efficiency of the proposed scheme is much more than the existing scheme.

Figure A2 shows the user registration in the proposed system. This user registration phase requires the user details based on user ID, user name, phone and email which will be stored in the database. Based on the above features we check the data integrity between the owner and user. Soon after the user registration, the system will provide an authentication to the data owner and in turn he or she can upload or choose the data in which they need to perform encryption as shown in figure A3. The data owner hides the data based on the DES encryption algorithm.

Once the encryption is completed, the encrypted data along with the access control list and mediator keys are uploaded to the cloud as in figure A4. Thus the cloud acts as a storage space. Then the user uses the private key and partially generated key to fully decrypt the data as shown in figure A5.

The efficiency of both the encryption and decryption operations is compared and plotted using bar graphs in figure 1. On comparing the encryption and decryption operations of existing scheme and proposed scheme, the efficiency of both these operations was found to be improved. The efficiency is improved by 5% in both the cases. X axis shows both encryption and decryption for existing and proposed system respectively. Y axis shows the corresponding efficiency rate against the X axis parameters. The bar graph result shows that the proposed system has better efficiency rate compared to the existing one. As the proposed scheme doesn't use the pairing operations, the encryption and decryption operations perform much more efficiently. The encryption time comparison based on message length is shown in figure 4. The encryption time increases linearly with increase in message size. Let the message size be 15Kb. The time taken to process the above message for the proposed and existing schemes will be 3ms and 6ms respectively as shown in figure 4. Also more users are allowed to access the same data item at the same time.

Figure 2 represents the system performance evaluation of the existing and proposed scheme based on the CPU architecture of the machine, storage, reliability and redundancy. Here the performance of the existing and the proposed systems are compared on the basis of the time required for performing both the encryption and decryption operations. Evaluation results show the system performance during the increase in number of processes in a machine. The graph depicts a

small variation while using the proposed system in a real time application. Figure 3 shows the performance evaluation on the basis of different message sizes. It can be seen from the graph that performance of the system increases with decrease in message size or data. For example consider the key length as 25Kb. After analyzing the efficiency with respect to the above key length, results improved slightly to 60 percent for the proposed one (50 percent for existing) in number of cycles per data and time. The cost of the existing scheme is high since the encryption algorithm is executed for each user. But the proposed scheme reduces cost since the data owner encrypts each data item with the encryption algorithm only once using a random symmetric key.

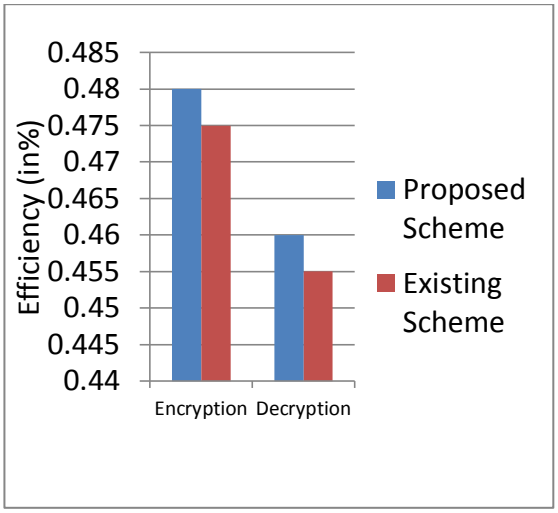

Figure 1.Comparison of encryption and decryption

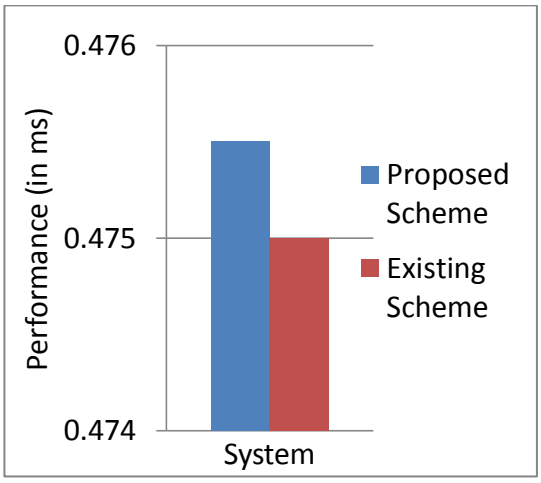

Figure 2.System performance evaluation

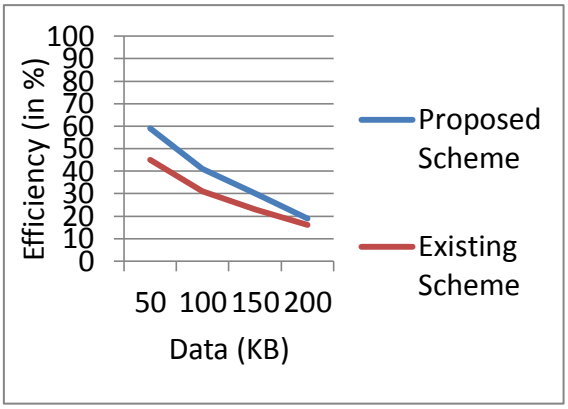

Figure 3.Performance comparison based on data

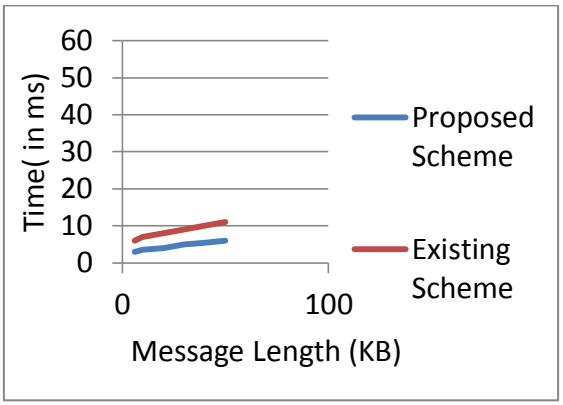

Figure 4.Encryption time comparison based on message length

6. CONCLUSION AND FUTURE WORK

In this paper**,** the security is improved by combining the data hiding schemes along with cryptographic schemes. From the graphs shown in the results and discussion section it is clear that the proposed scheme gives a slight improvement in the efficiency of both the encryption and decryption operations. In order to improve the efficiency of the system, once the initial partial decryption is performed for each user, the SMS stores the partially decrypted data in the cloud storage center. Thus partial decryption at the SMS reduces the load on users. The system is also cost effective and assures the confidentiality of the data stored in an untrusted public cloud. It improves data overhead using single encryption on each data item. In order to improve the searching capability of data items at the querying stage an enhanced encryption scheme should be added with the existing system and is considered for our future studies.

REFERENCES

[1] G.Miklau and D.Suciu, Controlling Access to Published Data Using Cryptography, Proceedings of the 29th international conference on Very large data bases VLDB, Berlin, Germany, 2003, pp. 898–909.

[2] M.Nabeel, N.Shang and E.Bertino, Privacy Preserving Policy Based Content Sharing in Public Clouds, IEEE Transactions on Knowledge and Data Engineering, Vol. 25, No. 11, 2012, pp. 2602–2614.

[3] S.Yu, C.Wang, K.Ren and W.Lou, Attribute Based Data Sharing with Attribute Revocation, Proceedings of the 5th ACM Symposium on Information, Computer and Computer and Communication Security, *IACCS*, New York, NY, USA, 2010, pp. 261–270, http://dx.doi.org/10.1145/1755688.1755720.

[4] S.Al-Riyami and K.Paterson, Certificateless Public Key Cryptography, 9th International Conference on the Theory and Application of Cryptology and Information Security ASIACRYPT, Taipei, Taiwan, 2003, pp. 452–473, http://dx.doi.org/10.1007/978-3-540-40061-5_29.

[5] X.W.Lei Xu and X.Zhang, CL-PKE: A Certificateless Proxy Re-Encryption Scheme for Secure Data Sharing with Public Cloud, Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security, Seoul, Republic of Korea, 2012, pp. 87-88, http://dx.doi.org/10.1145/2414456.2414507.

[6] S.S.M.Chow, C.Boyd and J.M.G.Nieto, Security Mediated Certificateless Cryptography, 9th International Conference on Theory Practice in Public-Key Crytopraphy, New York, NY, USA, 2006, pp. 508–524, http://dx.doi.org/10.1007/11745853_33

[7] Y.Sun, F.Zhang and J.Baek, Strongly Secure Certificateless Public Key Encryption without Pairing, 6th International Conference (CANS), Singapore, 2007, pp. 194–208, http://dx.doi.org/10.1007/978-3-540-76969-9_13.

[8] C.Yang, F.Wang and X.Wang, Efficient Mediated Certificates Public Key Encryption Scheme without Pairings, 21st International Conference on Advanced Information Networking and Applications Workshops, Ontario, Canada, 2007, pp. 109–112.

[9] Seung-Hyun Seo, Mohamed Nabeel, Xiaoyu Ding and Elisa Bertino, An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds, IEEE Transactions on Knowledge and Data Engineering, Vol. 26, No. 9, pp. 2107–2119, 2014, http://doi.ieeecomputersociety.org/10.1109/TKDE.2013.138.

[10] Z.Ni, Yung-Q.Shi, N.Ansari and W.Su, Reversible Data Hiding, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 16, No. 3, 2006, pp. 354–362.

[11] M.U.Celik, G.Sharma, A.M.Tekalp and E.Saber, Lossless Generalized-LSB Data Embedding, Journal of *IEEE* Transactions on Image Processing, Vol. 14, No. 2, 2005, pp. 253–266, http://dx.doi.org/10.1109/TIP.2004.840686.

[12] X.Zhang, Reversible Data Hiding in Encrypted Image, IEEE Signal Process Letters, Vol. 18, No. 4, 2011, pp. 255–258.

[13] X.Zhang, Separable Reversible Data Hiding in Encrypted Image, IEEE Transactions on Information Foresics and Security, Vol. 7, No. 2, 2012, pp. 826-832.

[14] M.Johnson, P.Ishwar, V.M.Prabhakaran, D.Schonberg and K.Ramchandran, On Compressing Encrypted Data, IEEE Transactions on Signal Processing, Vol. 52, No. 10, 2004, pp. 2992-3006.

[15] V.Suresh and C.Saraswathy, Separable Reversible Data Hiding Using Rc4 Algorithm, International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME), Salem, 2013, pp. 164 - 168.

[16] Akash Kumar Mandal, Chandra Parakash, Mrs.Archana Tiwari

Performance Evaluation of Cryptographic Algorithms: DES and AES, IEEE Transactions on Electrical, Electronics and Computer Science, Bhopal, 2012, pp. 1-5.

[17] Y.Vidya, B.Shemimol, Secured Friending in Proximity based Mobile Social Network, Journal of Excellence in Computer Science and Engineering, Vol.1, No. 2, 2015, pp. 1-10, http://dx.doi.org/10.18831/djcse.in/2015021001

[18] Announcing the Advanced Encryption Standard(AES), csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[19] X.Zhang, Lossy Compression and Iterative Reconstruction for Encrypted Image, IEEE Transactions on Information Forensics Security, Vol. 6, No. 1, 2011, pp. 53–58.

[20] P.Kadam, M.Nawale, A.Kandhare and M.Patil, Separable Reversible Encrypted Data Hiding in Encrypted Image Using AES Algorithm and Lossy Technique, International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME), Salem, 2013, pp. 312 - 316.

[21] V.Agham and T.Pattewar, A Novel Approach towards Separable Reversible Data Hiding Technique, International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014, Ghaziabad, India, pp. 771–775.

APPENDIX A

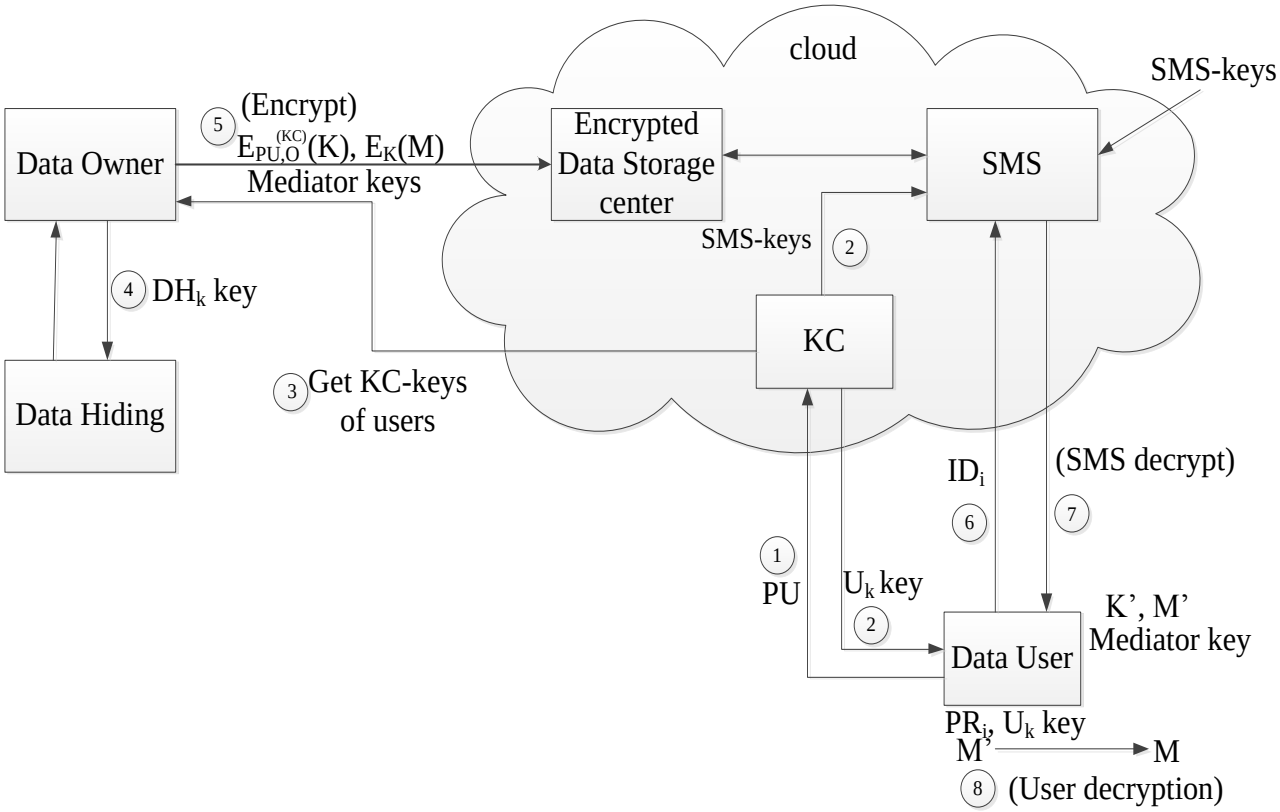

Figure A1.System overview

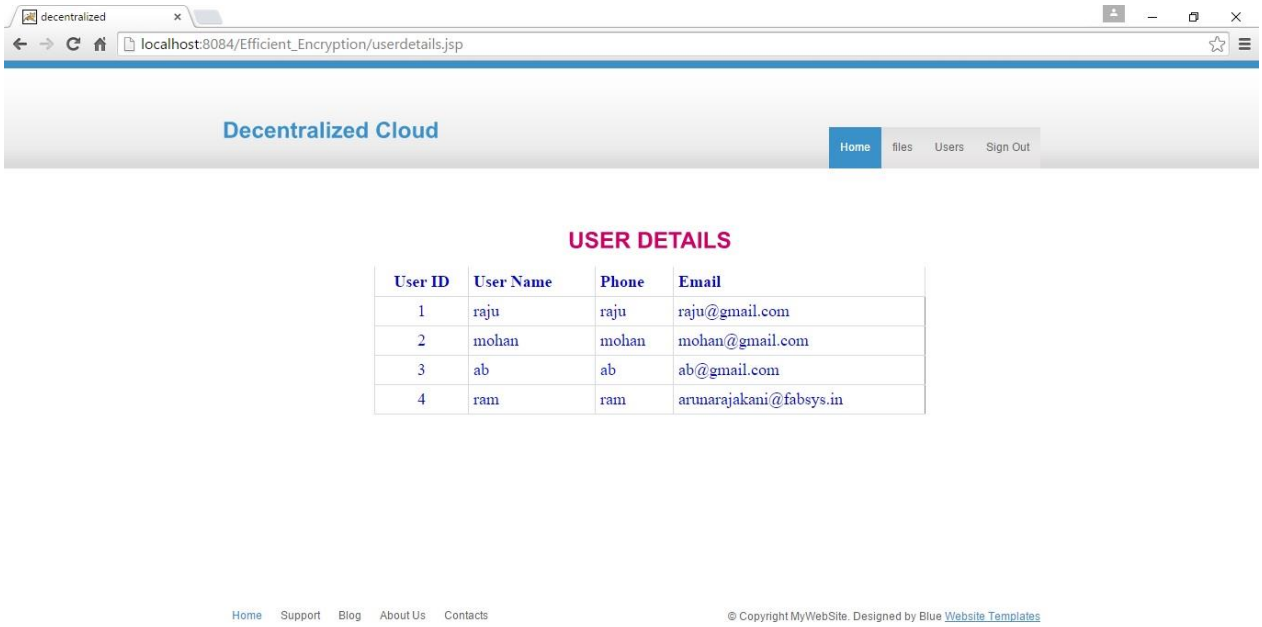

Figure A2.User registration

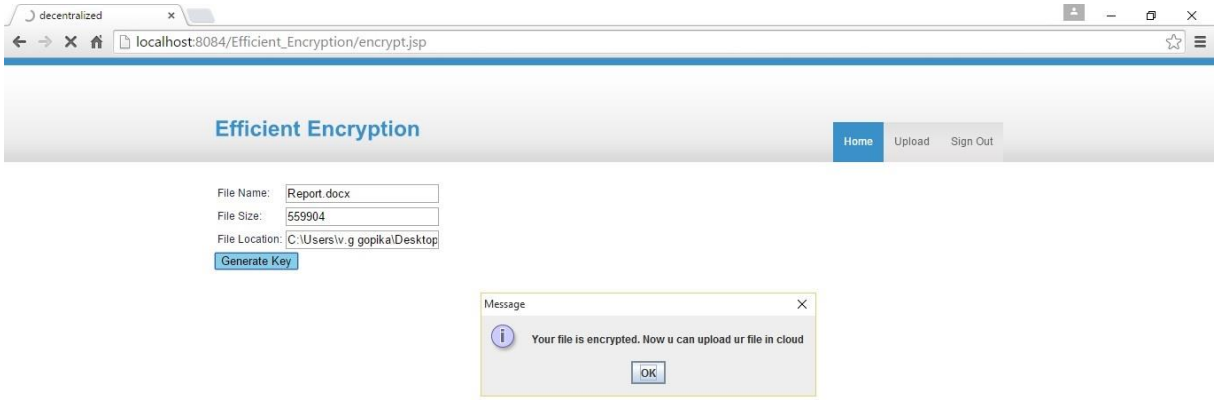

Figure A3.Data encryption with steganography

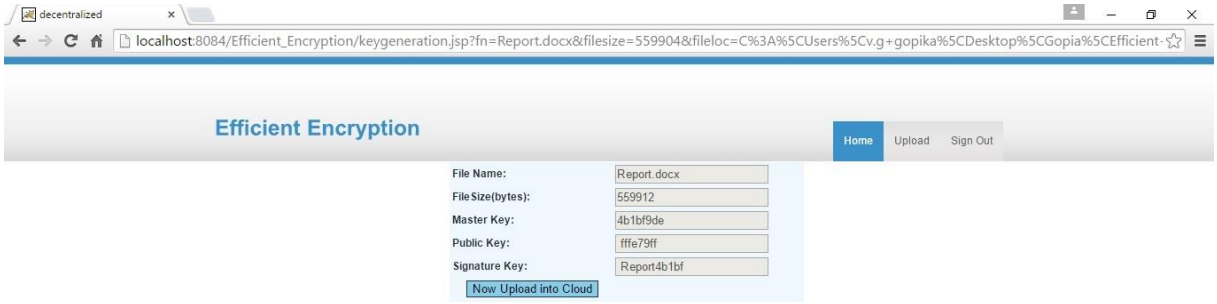

Figure A4.Data uploading

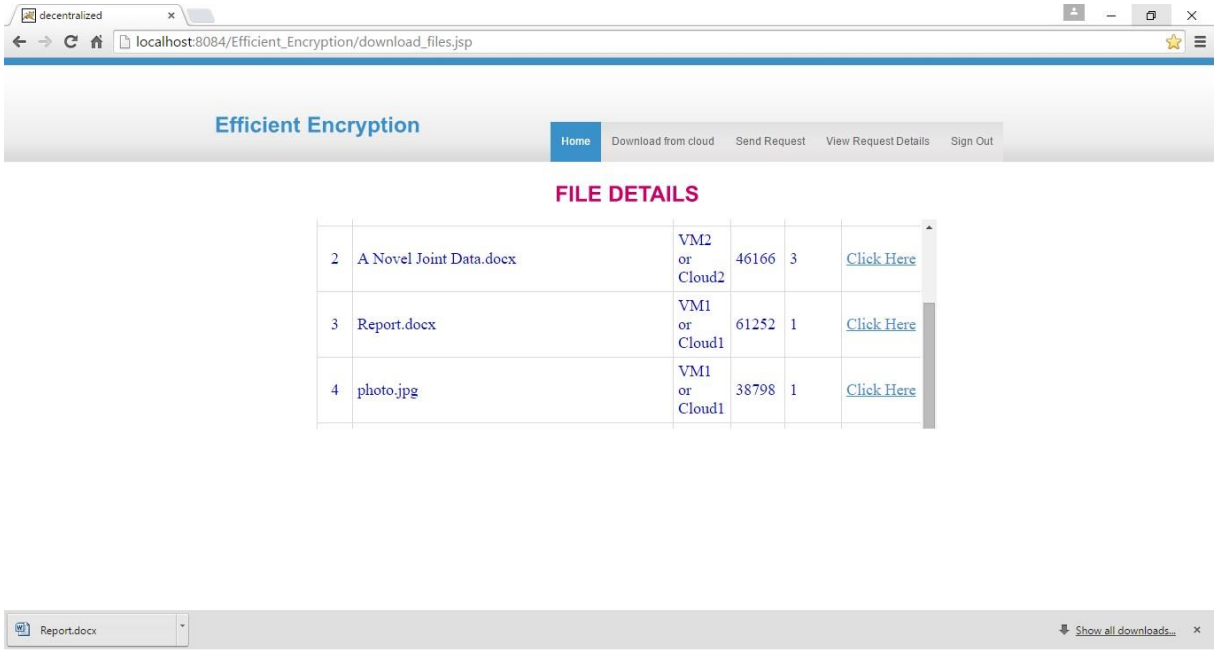

Figure A5.Data retrieval and decryption