ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

Enhanced Key Aggregation Technique for Secured Data Sharing in Cloud

Shweta. P. Tenginkai[1], Vani K. S[2]

Student, Department of CSE, Acharya Institute of Technology, Bangalore, India[1]

Assistant Professor, Department of CSE, Acharya Institute of Technology, Bangalore, India[2]

Abstract: Data sharing is a crucial functionality in cloud storage. With the current wireless technology and the usage of different smart devices, the sharing of data has become very easy with the help of Internet. Data sharing has become an important aspect to be considered in cloud. The cloud users want their files, folders, pictures and other confidential things to be available for their utilization. It is essential to safely, productively, and adaptably impart information to others in distributed storage. A special type of public-key encryption which is called enhanced Key-Aggregation Cryptosystem is designed to outline an effective scheme that helps in flexible delegation by a constant-size decryption key. The novelty is that one can aggregate any set of secret keys and make them as compact as a single key, but encompassing the decryption power of all the keys being aggregated. In modern cryptography, a fundamental problem is leveraging the secrecy of a small piece of knowledge into the ability to perform cryptographic functions multiple times. Here the efforts are taken for making the powerful decryption key, which permits decryption of different ciphertexts without expanding the aggregate key size. This scheme also holds well in the standard model and can be applicable anytime in cloud storage.

Keywords: Cloud storage, data sharing, ciphertexts, powerful aggregate key.

I. INTRODUCTION

Cloud computing is the new trending model used for computing in which the internet is used for communicating and storing the data. Some of the most crucial functionalities of cloud computing is data sharing and securely storing the important data dumped into cloud. When it comes to sharing and storing of data, the users of the cloud become bit hesitant to put the data onto the cloud scaring about the confidentiality and security of the data. Due to these aspects of preserving the security and confidentiality of the data, the notion of encryption came into picture. Here the users can encrypt their data using various encryption algorithms before putting them into the cloud. The users can also take the help of the Third Party key generators for encrypting and decrypting of data or can encrypt by themselves using various algorithms.

Cloud storage is day-by-day gaining popularity. It is being utilized as core technology for various online services. In today's world users may apply for free accounts for data sharing, emails, and storing confidential information with storage size up to 25 GB. The wireless technology enables us to access almost all the files, emails and data for the users using their smart devices from any remote corner of the world. Data sharing is a prime functionality in the cloud storage. The blog writers usually allow their friends to have a look or access some of the confidential pictures among the various pictures dumped in the cloud; any organization may grant their employees to access a small part of their confidential data. So here the sharing of the encrypted data with only the authentic users, who are given the rights to access it, is the challenging factor. Although users have the option of downloading the encrypted data from the cloud, decipher them, and later send them to their friends for sharing it, but this will simply lessen the impact of cloud storage. Instead the authentic users must be given the privilege of rights for accessing while data sharing with others in such a way for accessing those data directly from the server.

Cloud Storage is a service where data is remotely maintained, managed, and backed up. This service is available to users over a network, which is usually the internet. It allows the user to store files online so that the user can access them from any location via the internet [4].

The cloud concept that has recently become the technological hot topic is actually very old. It has roots dating back to the 1950's and 1960's. Computer scientist John McCarthy has been credited as one of the founding fathers of the cloud computing concept.

Cloud storage is a subcategory of the very complex cloud computing idea. It is a service model in which data is: maintained, managed and backed up remotely and made available to users over a network (typically the Internet). FilesAnywhere.com was one of the first companies to offer the cloud storage service. Their cloud storage service enables users to store data on their servers from anywhere at any time, while also being able to retrieve the data from anywhere at any time. FilesAnywhere.com would be a pioneer in the cloud storage business and many companies would follow suit [6].

Data sharing functionality is important in cloud storage. Consider Alice has some data to be store in the cloud and does not want expose it to anybody. She first encrypts the data and then uploads in the server in order to avoid data leakage [10] [16]. If Bob wants some data of the Alice then he requests her to share the data. Now the main task is sharing of the encrypted data. There are ways to do this.
1) Alice can encrypt the data using single and share the same key with Bob. 2) Alice can encrypt the data with

DOI 10.17148/IJARCCE.2015.4730

different keys and then send Bob's key to Bob using secure channel.

In the first approach, the data that is not required to be exposed may also get exposed to the bob while the second approach the numbers of keys required increases as the number of files and the number of users want to share the data. The storage space required to store the key and secure channel to share it also becomes expensive.

Encrypting the data using the different keys by the Alice and sending the single key for the decryption of the constant size to the Bob is the best solution. The decryption key has to be sent through the secure channel and the size of the secret key is smaller and enviable. The public key encryption scheme is supportable and is flexible such that any encrypted data is decrypt able by constant sized decryption key. Here we face some problem that how the data is efficiently stored in cloud computing. The user directly upload the data into cloud using the drop box without encryption, so the attacker can easily attack and it leads to missing data integrity and provides less security [2].

In the proposed method the data owner generates the public key after the account is created. He encrypts both the data and the public key upload in the cloud. The data owner also generates the aggregation detection key (ADK) using the public key. Data owner generates Aggregate Decryption Key (ADK) using its Public key [9] [13]. Data owner can share the data to other users by sending its ADK to those via Secured E-mail. After the verification of ADK, the other person can download the original data [15] [7] [8].

Data owner shares both the selected file and the ADK in order to download original data [4] [7]. Authentication of the file with ADK in the remote cloud ensures security. It provides confidentiality and data integrity [9] [11] [12] [14].

The encryption of the data came into picture to secure the confidential data of the user. This encryption involves generating and sharing of the secret keys. These secret keys' size varies for the different encryption schemes based on the length of files. There is a need for keeping the key size constant. The use of single aggregate key for decrypting data will avoid the burden of sharing many single secret keys. The aggregate key size should remain constant so that the network overhead can be reduced. This will also reduce the fuss of sharing many keys for deciphering the encrypted files.

As more and more enterprises and people are moving to cloud it is becoming crucial for people to get the security for the privacy of their data. Since data is a very essential thing to be secured and taken care, the data needs to be encrypted prior to putting onto the cloud. The aggregate key can be generated for any number of files that are dumped into the cloud. This aggregate key size remains constant using which the decryption of those files is possible for which the rights to access are given.

With the property of lower maintenance cost, cloud computing also makes it possible to share the files, data and other crucial information among cloud users. Unfortunately, preserving the privacy of the data from an un-trusted cloud is still a challenging issue. This project proposes a secure data privacy scheme, for user data in the cloud. The proper verification and validation of the user is done before uploading or downloading of the files into or from the cloud. The sharing of secret keys is done in a secured way by sending them in encrypted format through the e-mails.

A. *Need for Key Aggregation*

Key aggregation plays a prime role in overcoming the network overheads. The usage of different systems, smart phones, embedded systems and various devices, has increased the traffic on networks. On considering a scenario, where a particular user Alice wants to send an access key to her friend Bob, who wants to access some of the files. Alice has encrypted those files before uploading them onto the cloud. Then Alice can send an aggregate key of these corresponding secret keys of the various files using which Bob can decrypt them. Here, the load on network traffic is lowered, as the problem of sending all the corresponding keys is replaced by sending just a single aggregate key. The expenses of having a tamper proof storage are usually high. The cost of secured storage for storing these secret keys is also reduced by storing the aggregate key due to its compact size.

II. LITERATURE SURVEY

In this paper, we discuss the importance of bilinear maps in cryptography. They are important for applications like key exchange, encryption, and signatures. We discuss the use of bilinear maps to construct short signatures on elliptic curves, which is used as a foundation for the construction of compressed and aggregated signatures. From aggregation, we can also construct verifiably encrypted signatures as well as ring signatures. Bilinear maps are a very powerful tool, which might have many more undiscovered uses in cryptography.

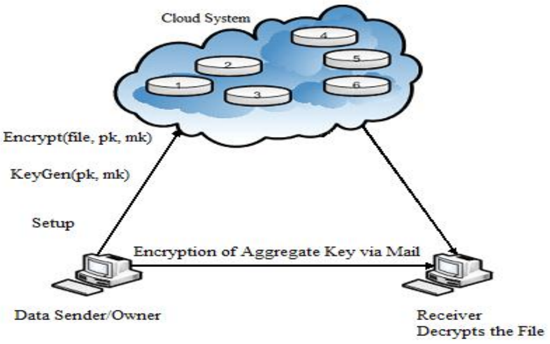

Fig. 1. The Proposed System Framework for Efficient Key Aggregation

The formal security is provided for the analysis of the schemes in the standard model for avoiding the unavoidable privilege access. To avoid the limitations in existing system, a new scheme of key-aggregation is proposed. In this scheme, the files or data to be put onto the cloud are first encrypted to preserve the data confidentiality and privacy. The new public-key encryption system gives ciphertexts of constant size in a

DOI 10.17148/IJARCCE.2015.4730

manner that there is better assignment of decryption rights given for larger number of ciphertexts. Data owner establishes the general public system parameter using the Setup for generating the public and private key combination using the KeyGen. The secret file is encrypted by using data encryption standard (DES) algorithm. The data owner will use the master-secret to come up with aggregate decipherment key for a collection of data files. The generated keys may be passed to delegates securely (through secure e-mails). Thus, any valid users with aggregate key will decrypt the data file and download it.

III.METHODOLOGY

The aggregation cryptosystem consists of efficient Key Aggregate Cryptosystem algorithm. The data owner set up the general public parameter using Setup and creates a public/private key and combines using KeyGen. The secret file is encrypted utilizing DES algorithm. The information owner will make use the master-secret to come up with aggregate decipherment key for a collection of data files. The generated keys may be passed to delegates securely (via secure e-mails or secure devices). Finally, any user with aggregate key will decrypt the data file and download it.

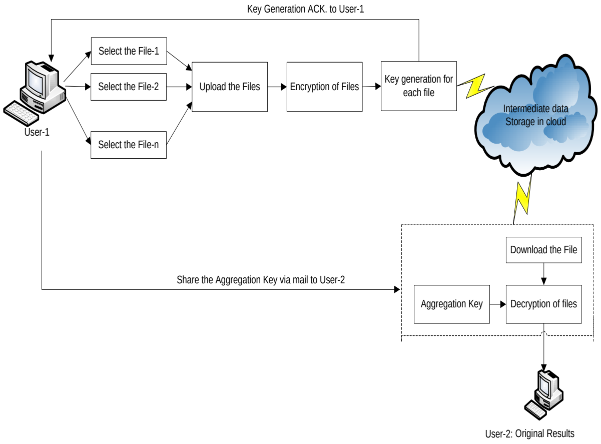

Fig. 1. System Architecture

Fig. 1. shows the structural behaviour of system. In this architecture, the scenario of two users is taken as an example, where the user-1 wants to upload the records onto the cloud, whereas the user-2 wants to download the records from the cloud. When the user-1 is uploading the records or the files, the data is first encrypted using the DES algorithm and the record gets uploaded onto the cloud. The generated respective private key for each file is displayed as an acknowledgement to the user-1. When the user-2 wants to view or access some files of user-1, he requests the user-1 to share the aggregate key of those particular files, using which after downloading the encrypted files is deciphered using that constant size aggregate key. The user-2 can now download and view all those files using the aggregate key.

An efficient Key-aggregate cryptosystem [1] produce constant size cipher texts such that efficient delegation of decryption rights for any set of cipher text are possible. During registration we have to create the public key and then login to the specific page. Here we can upload the files and also download the same files, if we are the valid user. If other person wants to access the data then they have to get the valid public key from the specific user and then they can access the decrypted data.

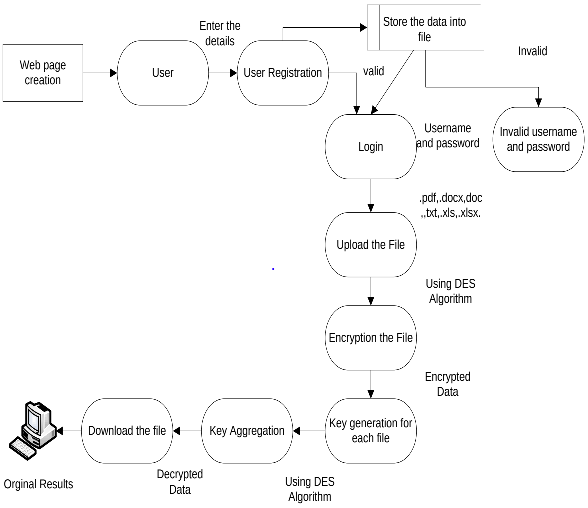

Fig. 2. Data Flow Diagram

The data owner establishes the general public system parameter via Setup and generates a public/private key and combine via KeyGen. The secret file is encrypted by using data encryption standard (DES). The data owner will use the master-secret to come up with aggregate decipherment key for a collection of data files. The generated keys may be passed to delegates securely (via secure e-mails or secure devices). Finally, any users with aggregate key will decrypted the data file and download the file.

It has the efficient key aggregation technique which has the uploading, key aggregate generator and downloading functionality. In upload module, the sender can upload number of files with private key. By using the combination of private and public key the data file is encrypted with the help of Data Encryption Standard (DES) algorithm and the encrypted data is stored into cloud. The efficient key aggregation technique is used to combine the private key and generate the fixed sized key called, the aggregate key. In the download module, the receiver can download the file by using the aggregate key which is mailed by the sender. If the aggregate key is valid, then the download for files is allowed. While the receiver is downloading the file, the aggregate key is verified or extraction of the keys is done along with those private keys. The decryption of the file is done by using DES algorithm.

The project is divided into the following modules based on the functionalities and operations.

New User Registration

This module contains the fields like Name, Username, Password, Email-id, Mobile number to be entered as a User. In this module all fields must be entered otherwise error message will be displayed. User must register with the cloud then perform the remaining operations .Without registration, other operations cannot be performed. So initially user registers and then goes for the login. In the user registration form user must enter the valid information otherwise user will get the error message, once the user registers by entering proper information it automatically generates message saying , the user successfully registered after completion.

Login
In the user login page, registered user must enter proper user name and password. After that, user performs the further operations. If any error occurs in the user name and password, an error message is displayed saying invalid username and password.

File Upload
Once the user logins successfully, user can upload the file. If the user selected option has an upload file, he then checks the extension of the file. Once user uploads it successfully a message is displayed saying you have successfully uploaded the file. Suppose user uploads a file now, user can encrypt a file by using encryption algorithm and after that the key is generated based on file name, user name, and this encrypted file and key are then stored in the cloud.

Key Generation Phase
Once the user uploads the file successfully, user can create key based on file using algorithms. Key will then be stored in the cloud and the filename will also be stored in the cloud system.

Requested User
In this module user can request to select the file and key for downloading the required files.

Requested Data View
In this module users will view the uploaded files and user can create key based on file using algorithm and the generated aggregate key will be sent through email.

Response User
In this module user will get the aggregate key through email and if aggregate key is matched it will verify the filename. Based on file if both values are true, then user can download the file.

File Download
If any other person wants to download the file uploaded by the file owner, he requests the key for particular file to the owner. The owner sends the aggregate key of the required files only to the person via email. The person uses this aggregated key to download the file. Then the file is decrypted using decryption algorithm. This decrypted file is then stored in the local system.

The (PBE) Password Based Encryption is a combination of hashing and symmetric encryption, where a 64-bit random number (the salt) is added to the password and hashed using the Message Digest Algorithm i.e. MD5 is used here.

Algorithm used

Input: Data files

Step 1: Setup- In this step the data owner or sender encrypts the data files (pdf,txt,doc). On input a security level parameter and the number of data file classes n (i.e., each file having private key should be generated by using public and private key pair), it outputs the public system parameter param, using Data encryption standard algorithms for security purpose.

Step 2: KeyGen- In this step the data owner/ sender randomly generate a public/master-secret key pair (pk, msk).

Step 3: Encrypt- In this step by using above steps the data owner/ sender encrypts the data file by using DES algorithm.The data file is encrypted by using hashing and symmetric key encryption. On input as a data file, a public-key and a private key, it outputs an encrypted data file.

Step 4:KAC- The data owner generates the aggregate key in aggregation cryptosystem by extracting the public-private key.

Step 5: Decryption- In this step who wants to download the list of data files using aggregate key, is sent by the data owner/sender to receiver's mail id directly. The receiver after receiving an aggregate key can download the list of data files from the cloud system.

Output: Downloaded file

IV. RESULTS AND DISCUSSIONS

In the project, the results and discussions are used for comparing the existing system with the proposed model by using the performance analysis plot as shown in Fig. 2. In this below figure, the performance analysis with existing and proposed system by comparing with compression factor and delegation ratio yields the more efficient result than the previous methods.

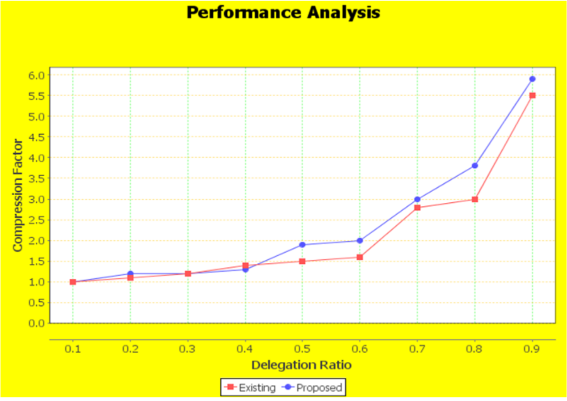

Fig. 2. Performance of the Proposed Work

V. CONCLUSION

In this paper, we discussed the public-key encryption methodology for protecting the privacy of data from the

DOI 10.17148/IJARCCE.2015.4730

attackers who may obtain the data by legal or other means, data stored by users and confidential information.

The main aim of this approach is to obtain the aggregate key of constant size empowered with the decryption rights for the number of files is possible by the valid user. Along with the privacy of data, the confidentiality is also preserved by encrypting the user data before dumping into the cloud.

Protection of the users' data privacy in cloud storage is an important aspect. With the help of mathematical tools, the encryption schemes are becoming more versatile and have started involving many encryption and decryption keys for a single application. But this project introduced the unique concept of the aggregation of the keys involved in decryption process. The cost of storing and transmitting the ciphertexts is lowered as they are constant-sized. This compact aggregate key can be conveniently sent to others or be stored in a smart card with very limited secure storage. It is modelled in such a way keeping different security levels and extensions.

Storing the delegated keys in the mobile devices which have no trusted software, there is a possibility that the key gets disclosed. So designing a leakage-proof cryptosystem which supports flexible and efficient key delegation is an interesting direction. In this project, the DES algorithm is used for encrypting the files. A more secured and efficient algorithm can be used in future so as to cope up with the speed and for security purpose. The measures to avoid data de-duplication in the cloud can also be one of the enhancements for this project.

REFERENCES

[1] Cheng-Kang Chu, S. M. Chow, Wen-G Tzeng, "Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage", Proc. IEEE Symp. Security and Privacy, Vol. 25, No. 2, Feb. 2014.

[2] K. Kate and S. D. Potdukhe, " Data sharing in cloud storage with key-aggregate cryptosystem", International Journal of Engineering Research and General Science, Volume 2, Issue 6, pp. 882-886, 2014.

[3] S. Prasanna and S. Ramya, "Implementation of Key aggregate Crypto with Stegnography for Secured Data Sharing in Cloud Computing", International Journal of Research in Computer Applications and Robotics, Volume 2, Issue 11, pp. 150-154, 2014.

[4] "Cloud storage", Nonprofit Technology Collaboration, 2013.

[5] T. Okamoto and K. Takashima, "Achieving Short Ciphertexts or Short Secret-Keys for Adaptively Secure General Inner-Product Encryption," Cryptology and Network Security , pp. 138-159, 2011.

[6] M. Evans, T. Huynh, K. Le and M. Singh," Cloud Storage", 2011.

[7] S. S. M. Chow, Y. Dodis, Y. Rouselakis and B. Waters, "Practical Leakage-Resilient Identity-Based Encryption from Simple Assumptions," Proc. ACM Conf. Computer and Comm. Security, pp. 152-161, 2010.

[8] M. Chase and S. S. M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," Proc. ACM Conf. Computer and Comm. Security, pp. 121-130. 2009.

[9] B. Alomair and R. Poovendran, "Information Theoretically Secure Encryption with Almost Free Authentication," J. Universal Computer Science, volume 15, Issue 15, pp. 2937-2956, 2009.

[10] Y. Sun and K. J. R. Liu, "Scalable Hierarchical Access Control in Secure Group Communications," Proc. IEEE INFOCOM '04, 2004.

[11] S.S.M. Chow, J. Weng, Y. Yang and R.H. Deng, "Efficient Unidirectional Proxy Re-Encryption," Proc. Progress in Cryptology, Volume 6055, pp. 316-332, 2010.

[12] D. Boneh, C. Gentry and B. Waters, "Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys," Advances in Cryptology Conference, Volume 3621, pp. 258-275, 2005.

[13] D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," Proc. Advances in Cryptology, volume 2139, pp. 213-229, 2001.

[14] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," ACM Trans. Information and System Security, Volume 9, Issue 1, pp. 1-30, 2006

[15] A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption," Theory and Applications of Cryptographic Techniques, Volume 3494, pp. 457-473, 2005.

[16] R. S. Sandhu, "Cryptographic Implementation of a Tree Hierarchy for Access Control," Information Processing Letters, Volume 27, Issue 2, pp. 95-98, 1988.

DOI 10.17148/IJARCCE.2015.4730