Secure Data Sharing in Cloud Computing Using Revocable-Storage Identity-Based Encryption

Swapnil Deshmukh, Sourabh Dhivare, Prof.Mr. Harshad Dagade

Abstract— with the growth of cloud computing, a new way for data sharing is being formed benefiting every other individual or organisation associated with it. But the data shared may not reach the cloud computing sources securely. This is when cryptography comes into picture to secure user's valuable data and information that needs to be kept thoroughly confidential. The authorization to secured data needs to be dynamic as soon as the authorization gets expired and needs to be revocable. This is where we propose (RS-IBE) where authorization are updated systematically and simultaneously. The performance of the proposed (RS-IBE) has advantages in terms of functionality and efficiency. Finally, we provide the implementation of the the proposed scheme as well as its practicability.

Index Terms— Cloud computing, data sharing, revocation, Identity-based encryption, cipher text update, decryption key exposure.

1 Introduction

Cloud computing is a paradigm that provides massive computation capacity and huge memory space at a low cost. It enables users to get intended services irrespective of time and location across multiple platforms (e.g., mobile devices, personal computers), and thus brings great convenience to cloud users. Among numerous services provided by cloud computing, cloud storage service, such as Apple's iCloud, Microsoft's Azure and Amazon's S3, can offer a more flexible and easy way to share data over the Internet, which provides various benefits for our society. However, it also suffers from several security threats, which are the primary concerns of cloud users. Firstly, outsourcing data to cloud server implies that data is out control of users. This may cause users' hesitation since the outsourced data usually contain valuable and sensitive information. Secondly, data sharing is often implemented in an open and hostile environment, and cloud server would become a target of attacks. Even worse, cloud server itself may reveal users' data for illegal profit. Thirdly, data sharing is not static. That is, when a user's authorization gets expired, he/she should no longer possess the privilege of accessing the previously and subsequently shared data. Therefore, while outsourcing data to cloud server, users also want to control access to these data such that only those currently authorized users can share the outsourced data.

A natural solution to conquer the problem is to use cryptographically enforced access control such as identity-based encryption (IBE).

1.1 Overview

The concept of identity-based encryption was introduced by Shamir, and conveniently instantiated by Boneh and Franklin. IBE eliminates the need for providing a public key infrastructure (PKI). Regardless of the setting of IBE or PKI, there must be an approach to revoke users from the system when necessary, e.g., the authority of some user is expired or the secret key of some user is disclosed. In the traditional PKI setting, several techniques are widely approved, such as certificate revocation list or appending validity periods to certificates. However, there are only a few studies on revocation in the setting of IBE. Boneh and Franklin first proposed a natural revocation way for IBE. They appended the current time period to the Cipher Text, and non-revoked users periodically received private keys for each time period from the key authority. Unfortunately, such a solution is not scalable, since it requires the key authority to perform linear work in the number of non-revoked users. In addition, a secure channel is essential for the key authority and non-revoked users to transmit new keys.

2 General System Description

Recently, Seo and Emura proposed an efficient RIBE scheme resistant to a realistic threat called decryption key exposure, which means that the disclosure of decryption key for current time period has no effect on the security of decryption keys for other time periods. Inspired by the above work and Liang et al. Introduced a cloud-based revocable identity-based proxy re-encryption that supports user revocation and ciphertext update. To reduce the complexity of revocation, they utilized a broadcast encryption scheme to encrypt the ciphertext of the update key, which is independent of users, such that only non-revoked users can decrypt the update key. However, this kind of revocation method cannot resist the collusion of revoked users and malicious non-revoked users as malicious non-revoked users can share the update key with those revoked users.

3 Methodology

We introduce a notion called revocable storage identity-based encryption (RS-IBE) for building a cost-effective data

sharing system that fulfills the three security goals. More precisely, the following achievements are captured in this paper:

• We provide formal definitions for RS-IBE and its corresponding security model.

• We present a concrete construction of RS-IBE. The proposed scheme can provide confidentiality and backward/forward secrecy simultaneously.

We prove the security of the proposed scheme in the standard model, under the decisional ℓ-Bilinear Diffie-Hellman Exponent (ℓ-BDHE) assumption. In addition, the proposed scheme can withstand decryption key exposure:

The proposed scheme is efficient in the following ways:

They utilized the idea to provide the forward secrecy of Cipher Text, rather than secret key as in the original case.

Our scheme achieves forward security under the assumption that the encrypted data is stored in the cloud and users do not store the encrypted/decrypted data locally.

– The procedure of Cipher Text update only needs / public information. Note that no previous identity-based encryption schemes in the literature can provide this feature;

– The additional computation and storage complexity, which are brought in by the forward secrecy, is all upper bounded by O (log (T) 2), where T is the total number of time periods.

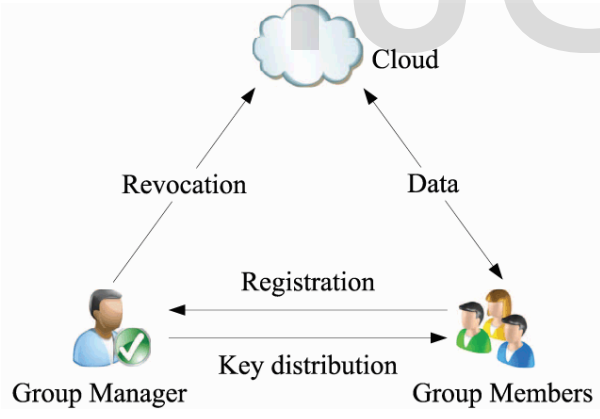


Fig 1.

4 Problem Definition and Solution

4.1 Problem Definition

The specific problem addressed in this paper is how to construct a fundamental identity-based cryptographically tool to achieve the above security goals. We also note that there exist other security issues that are equally important for a practical system of data sharing, such as the authenticity and availability of the shared data.

4.2 Problem Solutions

It seems that the concept of revocable identity-based encryption (RIBE) might be a promising approach that fulfils the aforementioned security requirements for data sharing. RIBE features a mechanism that enables a sender to append the current time period to the Cipher Text such that the receiver can decrypt the Cipher Text only under the condition that he/she is not revoked at that time period. As indicated in Figure 1, a RIBE-based data sharing system works as follows:

Step 1: The data provider (e.g., David) first decides the users (e.g., Alice and Bob) who can share the data. Then, David encrypts the data under the identities Alice and Bob, and uploads the Cipher Text of the shared data to the cloud server.

Step 2: When either Alice or Bob wants o get the shared data, she or he can download and decrypt the corresponding Cipher Text. However, for an unauthorized user and the cloud server, the plaintext of the shared data is not available.

Step 3: In some cases, e.g., Alice's authorization gets expired, David can download the Cipher Text of the shared data, and then decrypt-then-re-encrypt the shared data such that Alice is prevented from accessing the plaintext of the shared data, and then upload the re-encrypted data to the cloud server again.

Obviously, such a data sharing system can provide confidentiality and backward secrecy. Furthermore, the method of decrypting and re-encrypting all the shared data can ensure forward secrecy. However, this brings new challenges. Note that the process of decrypt-then-re-encrypt necessarily involves users' secret key information, which makes the overall data sharing system vulnerable to new attacks. In general, the use of secret key should be limited to only usual decryption, and it is inadvisable to update the cipher text periodically by using secret key. Another challenge comes from efficiency. To update the Cipher Text of the shared data, the data provider has to frequently carry out the procedure of download-decrypt-encrypt- upload. This process brings great communication and computation cost, and thus is cumbersome and undesirable for cloud users with low capacity of computation and storage. One method to avoid this problem is to require the cloud server to directly re-encrypt the Cipher Text of the shared data. However, this may introduce cipher text extension; namely, the size of the Cipher Text of the shared data is linear in the number of times the shared data have been updated. In addition, the technique of proxy re-encryption can also be used to conquer the aforementioned problem of efficiency. Unfortunately, it also requires users to interact with the cloud server in order to update the Cipher Text of the shared data.

5 FLOW OF WORKING

- **User Revocation**

An advanced version of user revocation has been proposed. By the revoked user modification of tags, their updated operations are potentially communication intensive. PA performs challenge algorithm during the auditing process of data where it is simultaneous modified by the revoked user. Finally the integrity of challenged file can be verified by the running verification algorithm.

- **Integrity Auditing**

The integrity auditing scheme for shared data on cloud based on ring signature based homophobic authenticates. In this scheme user revocation is not consider and the auditing cost grows with group size and data size enhanced their previous public integrity verification scheme with the support of user revocation. However if the cloud node responsible for tag update is compromised during user revocation process, attackers can discover the secret keys for all other valid users.

- **Cloud Storage**

Cloud stores file in cloud storage where the group admin also stores their file in cloud. Secret key is generated by a group admin and send to all the group members. The Group owner will send group key which will be common to all the group users and if the key is not been used within 48 hours the OTP will be generated.User details and secret key are stored in cloud storage. When user access the file the cloud storage access to allow the file to view and modify the content directly from the cloud storage.

- **Public Verification**

When user revocation occurs our scheme only required master user to send one group element to the cloud and also add owner group element to a public key. Public integrity auditing technique which is proposed in terms of efficient and data corruption detection probability.

- **Time Allocation**

To access the file from the cloud storage the user send a secrete key of a particular file third party authority (TPA) .TPA allocate the particular file modification time for the user. When the time will be expired the access time for the user so the time allocated for another user to access the same file within the particular time period given to them.

- **Third Party Authority**

The cloud server is the party that provide data storage service to the group user.The master user, who is the owner of the shared data and managers the membership of the other group user.The third party authority refers to a module that checks the integrity of data being stored on the cloud.

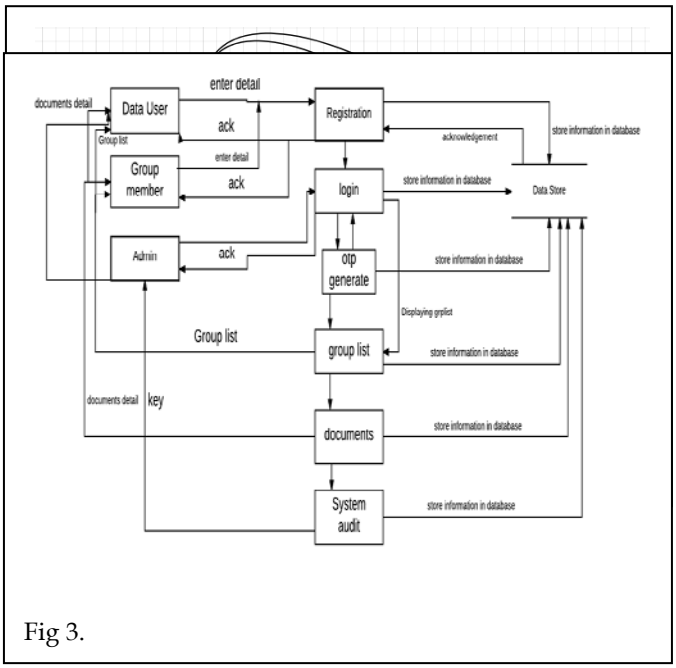
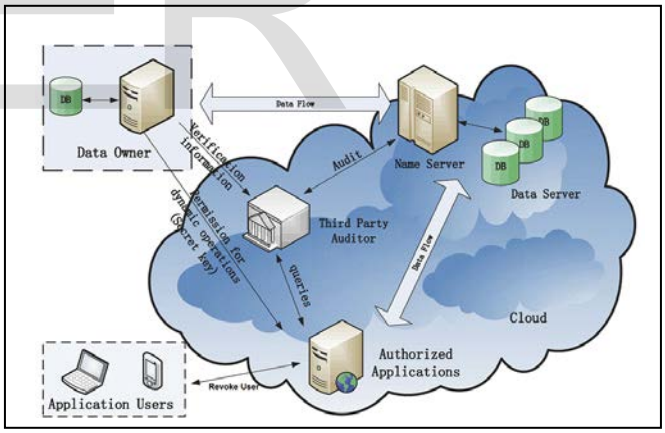


Fig 3.

5.1 Dataflow Diagram

5.2 System Architecture Diagram



6 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

6.1 Economic Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

6.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

6.3 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

7 Conclusion

Cloud computing brings great convenience for people. Particularly, it perfectly matches the increased need of sharing data over the Internet. In this paper, to build a cost-effective and secure data sharing system in cloud computing, we proposed a notion called RS-IBE, which supports identity revocation and Cipher Text update simultaneously such that a revoked user is prevented from accessing previously shared data, as well as subsequently shared data. Furthermore, a concrete construction of RS-IBE is presented. The proposed RS-IBE scheme is proved adaptive-secure in the standard model, under the decisional ℓ-DBHE assumption. The comparison results demonstrate that our scheme has advantages in terms of efficiency and functionality, and thus is more feasible for practical applications.

- *Swapnil Deshmukh is currently pursuing Bachelor's degree program in Information Technology engineering in Mumbai University, India. E-mail: deshmukhswapnil20994@gmail.com*
- *Sourabh Dhivare is currently pursuing Bachelor's degree program in Information Technology engineering in Mumbai University, India. E-mail: saurabh.dhivare@gmail.com*
- *Harshad Dagade is Assistant Professor in college in Mumbai University, India. E-mail: harshad.friend@gmail.com*

8 References

[1]. *L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," ACM SIGCOMM Computer Communication Review, vol. 39, no. 1, pp. 50–55, 2008.*

[2]. *iCloud. (2014) Apple storage service. [Online]. Available: https://www.icloud.com/*

[3]. *Azure. (2014) Azure storage service. [Online]. Available: http://www.windowsazure.com/*

[4]. *Amazon. (2014) Amazon simple storage service (amazons3). [Online]. Available:http://aws.amazon.com/s3/*

[5]. *K. Chard, K. Bubendorfer, S. Caton, and O. F. Rana, "Social cloud computing: A vision for socially motivated resource sharing," Services Computing, IEEE Transactions on, vol. 5, no. 4, pp. 551–563, 2012.*

[6]. *C. Wang, S. S. Chow, Q. Wang, K. Ren, and W. Lou, "Privacypreserving public auditing for secure cloud storage," Computers, IEEE Transactions on, vol. 62, no. 2, pp. 362–375, 2013.*

[7]. *G. Anthes, "Security in the cloud," Communications of the ACM, vol. 53, no. 11, pp. 16–18, 2010.*

[8]. *K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," Parallel and Distributed Systems, IEEE Transactions on, vol. 24, no. 9, pp. 1717–1726, 2013.*

[9]. *B. Wang, B. Li, and H. Li, "Public auditing for shared data with efficient user revocation in the cloud," in INFOCOM, 2013 Proceedings IEEE. IEEE, 2013, pp. 2904–2912.*

[10]. *S. Ruj, M. Stojmenovic, and A. Nayak, "Decentralized access control with anonymous authentication of data stored in clouds," Parallel and Distributed Systems, IEEE Transactions on, vol. 25, no. 2, pp. 384–394, 2014.*