RESEARCH ARTICLE OPEN ACCESS

Forward Secrecy and Data Sharing Through RSIBE in Cloud Computing

M. Sravani [1], Sk.Sofia [2]
M.Tech (CSE) [1], Asst. Professor [2]
Department of Computer Science and Engineering
Priyadarshini College of Engineering and Technology
Affiliated to JNTUniversity, Ananthapuram, Nellore
Andhra Pradesh - India

ABSTRACT

Cloud computing provides a flexible and convenient way for data sharing, which brings various benefits for both the society and individuals. But there exists a natural resistance for users to directly outsource the shared data to the cloud server since the data often contain valuable information. Thus, it is necessary to place cryptographically enhanced access control on the shared data. Identity based encryption is a promising cryptographically primitive to build a practical data sharing system. However, access control is not static. That is, when some user's authorization is expired, there should be a mechanism that can remove him/her from the system. Consequently, the revoked user cannot access both the previously and subsequently shared data. To this end, we propose a notion called revocablestorage identity based encryption (RSIBE), which can provide the forward/backward security of cipher text by introducing the functionalities of user revocation and cipher text update simultaneously. Furthermore, we present a concrete construction of RSIBE, and prove its security in the defined security model. The performance comparisons indicate that the proposed RSIBE scheme has advantages in terms of functionality and efficiency, and thus is feasible for a practical and costeffective datasharing system. Finally, we provide implementation results of the proposed scheme to demonstrate its practicability.

Keywords: Cloud computing, data sharing, revocation, Identity based encryption, cipher text update, decryption key exposure.

I. INTRODUCTION

Cloud computing is a paradigm that provides massive computation capacity and huge memory space at a low cost [1]. 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 [2], Microsoft's Azure [3] and Amazon's S3 [4], can offer a more flexible and easy way to share data over the Internet, which provides various benefits for our society [5], [6]. However, it also suffers from several security threats, which are the primary concerns of cloud users [7].

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 aforementioned problem is to use cryptographically enforced access control such as identity based encryption (IBE). Furthermore, to over come the above security threats, such kind of identity based access control placed on the shared data should meet the following security goals:

- **Data confidentiality**: Unauthorized users should be prevented from accessing the plaintext of the shared data stored in the cloud server. In addition, the cloud server, which is supposed to be honest but curious, should also be deterred from knowing plaintext of the shared data.
- **Backward secrecy**: Backward secrecy means that, when a user's authorization is expired, or a user's secret key is compromised, he/she should be prevented from accessing the plaintext of the *subsequently* shared data that are still encrypted under his/her identity.
- **Forward secrecy**: Forward secrecy means that, when a user's authority is expired, or a user's secret key is compromised, he/she should be prevented from accessing the plaintext of the shared data that can be *previously* accessed by him/her.

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 [8], [9], [10], [11], [12]. But the research on these issues is beyond the scope of this paper.

1.1 Motivation

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 Aliceor Bobwants to 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.

the cloud server to directly re encrypt the cipher text of the shared data. However, this may introduce cipher text exten sion, 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.

1.2 Related work

1.2.1 Revocable identity based encryption

The concept of identity based encryption was introduced by Shamir [13], and conveniently instantiated by Boneh and Franklin [14]. 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, the problem of revocation has been well studied [15], [16], [17], [18], [19], and 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 [14] 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. To conquer this problem, Boldyreva, Goyal and Kumar [20] introduced a novel approach to achieve efficient revocation. They used a binary tree to manage identity such that their RIBE scheme reduces the complexity of key revocation to logarithmic (instead of linear) in the maximum number of system users. However, this scheme only achieves selective security. Subsequently, by using the aforementioned revocation technique, Libert and Vergnaud [21] proposed an adaptively secure RIBE

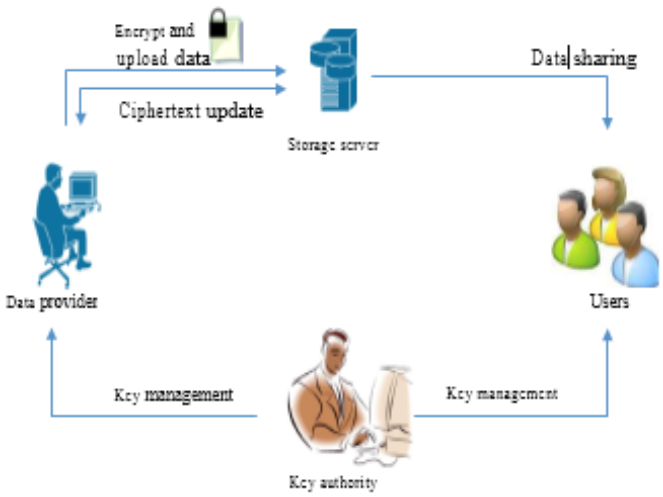

Fig. 1. A natural RIBEbased data sharing system

Obviously, such a data sharing system can provide con fidentiality and backward secrecy. Furthermore, the method of decrypting and re encrypting all the shared data can en sure forward secrecy. However, this brings new challenges. Note that the process of decryptthenreencrypt 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 downloaddecryptre encryptupload. This process brings great communication and computation cost, and thus is cumbersome and unde sirable for cloud users with low capacity of computation and storage. One method to avoid this problem is to require scheme based on a variant of Water 's IBE scheme [22], Chen et al. [23] constructed a RIBE scheme from lattices. Recently, Seo and Emura [24] 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 [25], Liang et al. [26] introduced a cloud based revocable identity based proxy re encryption that supports user revocation and cipher text update. To reduce the com plexity of revocation, they utilized a broadcast encryption scheme [27] to encrypt the

cipher text 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. Furthermore, to update the cipher text, the key authority in their scheme needs to maintain a table for each user to produce the re encryption key for each time period, which significantly increases the key authority's workload.

1.2.2 Forwardsecure cryptosystems

In 1997, Anderson [28] introduced the notion of forward security in the setting of signature to limit the damage of key exposure. The core idea is dividing the whole lifetime of a private key into T discrete time periods, such that the compromise of the private key for current time period cannot enable an adversary to produce valid signatures for previous time periods. Subsequently, Bellare and Miner provided formal definitions of forward secure signature and presented practical solutions. Since then, a large number of forward secure signature schemes [29], [30], [31], [32], [33] has been proposed.

In the context of encryption, Canetti, Halevi and Katz [34] proposed the first forward secure public key encryption scheme. Specifically, they firstly constructed a binary tree
encryption, and then transformed it into a forward secure encryption with provable security in the random oracle model. Based on Canetti et al's approach, Yao et al. [35] proposed a forward secure hierarchical IBE by employing two hierarchical IBE schemes, and Nieto et al. [36] designed a forward secure hierarchical predicate encryption.

Particularly, by combining Boldyreva et al.'s [20] revocation technique and the aforementioned idea of forward security[1], in CRYPTO 2012 Sahai, Seyalioglu and Waters [37] proposed a generic construction of so called revocable storage attribute based encryption, which supports user revocation and cipher text update simultaneously. In other words, their construction provides both forward and backward secrecy. What must be pointed out is that the process of cipher text update of this construction only needs public information. However, their construction cannot be resistant to decryption key exposure, since the decryption is a matching result of private key and update key.

– The procedure of cipher text update only need s *public information*. Note that no previous identity based encryption schemes in the literature can provide this feature;
– The additional computation and storage com plexity, 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.

Outline. The remainder of this paper is structured as fol lows: In section 2, we introduce the preliminaries involved in our construction. Then we present the definitions of RS IBE in section 3, and provide the concrete construction in section 4, followed with the corresponding security analysis, performance discussions, and the implementation results of the scheme. Finally, we conclude in section 5.

1.3 Our contributions

In this paper, we introduce a notion called revocable storage identity based encryption (RSIBE) 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 present a concrete construction of RSIBE. The proposed scheme can provide confidentiality and backward/forward[2] secrecy simultaneously;
- We prove the security of the proposed scheme in the standard model, under the decisional ℓBilinear Diffie Hellman Exponent (ℓBDHE) assumption. In addition, the proposed scheme can withstand decryption key exposure;
- The proposed scheme is efficient in the following ways:

1. They utilized the idea to provide the forward secrecy of cipher text, rather than secret key as in the original case.
2. As in [37], our scheme achieves forward security under the as sumption that the encrypted data is stored in the cloud and users do not store the encrypted/decrypted data locally.

$\mathbf{Adv}^{\ell-\mathrm{dBDHE}}$

2

We say that the decisional ℓBDHE assumption holds in G_1 *provided that no PPT algorithm can solve the decisional ℓBDHE problem with a non negligible advantage.*

II. PRELIMINARIE S

In this section, we first briefly present the basic concepts on bilinear pairing and decisional ℓBDHE assumption. Then, an algorithm used to perform efficient revocation is introduced.

2.1 Bilinear pairing and complexity assumption

Definition 1 (Bilinear pairing). *Let* G_1 *and* G_2 *be two cyclic groups with prime order* q, *and* g *be a generator of* G_1 *. A bilinear*
Pairing is a map $e : G_1 \times G_1 \rightarrow G_2$ *with the following*

properties:

G_1 *with prime order* p *according to the security parameter* λ.

2.2 KUNodes algorithm

Our RSIBE scheme uses the same binary tree structure introduced by Boldyreva, Goyal and Kumar [20] to achieve efficient revocation. To describe the revocation mechanism, we first present several notations. Denote by ε the root node of the binary tree BT , and Path(η) the set of nodes on the path from ε to the leaf node η (including ε and η). For a nonleaf node θ, we let θl and θr stand for its left and right child, respectively. Given a time period

t and revocations list RL, which is comprised of the tuples (ηi , ti) indicating that the node ηi was revoked at time period ti , the algorithm KUNodes (BT , RL, t) outputs the smallest subset Y of nodes of BT such that Y contains an ancestor for each node that is not revoked before the time period t.f t

$$\Pr C(f, D = e(g^{s}, g^{a})) = 0$$

$$= \mathbf{Adv}^{\text{IND-RID-CFA}}_{\text{RS-IBE, A}}(\lambda, T, N) + 2 \cdot \Pr[E].$$

However, if D is a random element from G2 then encrypted message M ∗ is perfectly hidden from the A's view, And thus

$$\Pr[C(f, D \xleftarrow{R} G) = 0] = \frac{1}{} \cdot \Pr[E].$$

III. PERFORMANCE DISCUSSIONS

In this section, we discuss the performance of the proposed RSIBE scheme by comparing it with previous works in terms of communication and storage cost, time complex ity and functionalities, which are summarized in Table 1, Table 2 and Table 3.

Utilize binary data structure to achieve revocation. On the other hand, Liang et al.'s [26] scheme involves a broadcast encryption scheme to distribute update key such that their scheme has constant sizes of private key and update key. Furthermore, by delegating the generation of re encryption key to the key authority, the cipher text size of their scheme also achieves constant. However, to this end, the key authority has to maintain a data table for each user to store the user 's secret key for all time periods, which brings O(T)τG z storage cost for the key authority. Conversely, the ciphertext size of our scheme is just linear in log(T)2 . In addition, we note that in all listed schemes, the private key generator needs to periodically produce an update key, it must be online if each time period is rather short, e.g., an hour. However, from the perspective of practical applications, the frequency of updating users' decryption keys should not be too small. A time period like a week, half a month or a moth is more desirable. As a consequence, the private key generator just needs to produce an update key for the next period when the current time period is over. Thus the PKG does not need to be always online. Another limitation of these listed schemes is that the generated ciphertext has the size linear with the number of receivers. To overcome this issue, a natural manner is to construct a similar scheme in the setting of broadcast encryption.

The time costs of the algorithms **CTUpdate** and **Decrypt**. is over. Thus the PKG does not need to be always online. Another limitation of these listed schemes is that the generated cipher text has the size linear with the number of receivers. To overcome this issue, a natural manner is to construct a similar scheme in the setting of broadcast encryption.

As shown in Table 3, the four schemes are all proved secure in an adaptive secure model, and can also provide backward secrecy since they all supports identity revocation. But the security of our scheme is built upon a relatively strong security assumption, decisional ℓDBHE assumption. The schemes [22], [24] and ours update user's secret keys in a public way, namely, the update key is available for all users. However, Liang et al.'s [26] scheme involves the method of broad encryption to update user 's secret key such that only non-revoked users can obtain the update key. Consequently, their scheme cannot resist collusion attack of revoked users and non revoked users. Compared with the schemes [22] and [24], Liang et al.'s [26] scheme and ours can both provide forward secrecy by additionally introducing the functionality of cipher text update. But the procedure of cipher text update in Liang et al.'s [26] scheme is performed in a private and interactive way, since it requires the key authority to periodically produce and provide re encryption keys for the cloud server to update cipher text. However, in our schemes, the cloud server itself can update cipher text by just using public parameter.

IV. IMPLEMENTATION

To show the practical applicability of the proposed RS IBE scheme, we further implement it using codes from the Pairing Based Cryptography library version 0.5.14 [39]. Specifically, we use the symmetric super singular curve $y^2 = x^3 + x$, where the base field size is 512bit and the embedding degree is 2. The implementation is taken on a Linux like system (Win7 + Min GW) with an Intel(R) Core(TM) i5 CPU (650@3.20GHz) and 4.00 GB RAM.

In the implementation, we set the number of users to be $N = 8$ and the revoked uses to be $R = 4$ (the nodes η_2 , η_3 , η_4 , η_7 in Figure 2 are revoked). In Figure 5, Figure 6 and Figure 7, we present the running time of the basic algorithms, i.e., **PKGen**, **KeyUpdate**, **DKGen**, **Encrypt**, **CTUp date** and **Decrypt**, for different choice of the total number of time periods $T \in \{2^4, 2^6, 2^8, 2^{10}, 2^{12}, 2^{14}, 2^{16}, 2^{18}\}$. To gen$_{\text{erate the}}$

experimental results, we perform as the following procedure: generate the private key and encrypt a message at the initial time period, then, periodically update the private key and the cipher text, and decrypt the cipher text. For a small number of time periods: $T \in \{2^4, 2^6, 2^8\}$, the running time of each algorithm is obtained by computing the average of running the above procedure 100 times. While, for a large number of time periods: $T \in \{2^{10}, 2^{12}, 2^{14}, 2^{16}, 2^{18}\}$, the running time for each algorithm is obtained by running the above procedure only once, and the running time for update algorithm is the mean of the first 512 time periods. We observe that, the time costs of the algorithms **PKGen**,

6. In our scheme, given the decryption $DK_{ID,t}$ and cipher text $CT_{ID,t'}$, if $t \geq t'$ then the cloud server would update $CT_{ID,t'}$ to $CT_{ID,t}$. Here, we just consider the decryption complexity for an individual **Key Update**, **DK Gen** and **Decrypt** are independent of the total number of time periods, and no more than 40 millisecond On the other hand, it takes less than 1 second for the user to initially encrypting the message, which would be share on the cloud. Although the time cost of the algorithm **CT Update** is apparently greater than other algorithms, it is run by a cloud server with powerful capability of computation. Thus, our RSIBE scheme is feasible for practical applications.

V. CONCLUSIONS

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 RSIBE, 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 RSIBE is presented. The proposed RSIBE scheme is proved adaptive secure in the standard model, under the decisional ℓDBHE assumption. The comparison results demonstrate that our scheme has advantages in terms of efficiency and functionality, and thus is more feasible for practical applications.

ACKNOWLEDGEM ENTS

We thank the anonymous reviewers for their helpful comments and suggestions.

REFERENCES

[1] L. M. Vaquero, L. RoderoMerino, 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 (amazon s3). [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, "Privacy preserving public auditing for secure cloud storage," *Computers, IEEE Transactions on*, vol. 62, no. 2, pp. 362–375, 2013.

Authors

Ms.M.Sravani has received B.Tech degree in Computer Science from PBR Vits,Kavali affiliated to JNTU Anathapuram in the year 2014 and Pursuing M.Tech in Computer Science and Engineering(CSE) from Priyadarshini College of engineering and Technology(Affiliated to NTUniversity, Ananthapuram), Nellore, Andhra Pradesh, India.

Assistant Prof.Sk.Sofia has received B.Tech degree in Computer Science and Engineering from Priyadarshini College of Engineering, Nellore and M.Tech in Computer Science and Engineering from Audi Sankara College of Engineering and Technology Nellore, affiliated to JNTUnversity, Ananthanpuram, Her interesting domain is Networks Security, she attended 2 National Conferences and published 3 papers in various International Journals. She is dedicated to teaching field from last 6 years, presently working as Assistant Professor in Computer Science and Engineering Department at Priyadarshini College of Engineering and Technology (Affiliated to JNTUniversity, Ananthapuram), Nellore, Andhra Pradesh, India