International Journal of
Scientific Engineering and Technology Research

ISSN 2319-8885
Vol.06,Issue.32
November-2017,
Pages:6571-6577

www.ijsetr.com

Proxy Re-Encryption Scheme for Data Security in Cloud

BOGGULA ARISTOTLE[1], V.N.V. REVANTH KUMAR[2]

[1]PG Scholar, Dept of CSE, Sreenivasa College of Engineering and Technology, Kurnool, AP, India,
Email: sayhai2socrates@gmail.com.

[2]Associate Professor, Dept of CSE, Sreenivasa College of Engineering and Technology, Kurnool, AP, India,
Email: vnvrevanth1988@gmail.com.

Abstract: A Cloud storage system consists a collection of storage servers and provides a long term storage services over the internet. Storing the data in a third party's cloud causes a problem with the data confidentiality. Efficient Conditional Proxy Re-encryption formalizes its semantic security. It allows the sender to encrypt data/message to multiple receivers by providing the receivers' identities. Then the sender sends a re-encryption key to the proxy, so that he/she can convert the cipher text into a new one to the intended receivers' new set. The re-encryption key is associated with a condition such that only the matching cipher texts can be re-encrypted. In a fine-grained manner,it allows the original sender to access control over the cipher texts.

Keywords: Proxy Re-Encryption, Cloud Storage, Identity-Based Encryption, Broadcast Encryption, Secure Cloud Email.

I. INTRODUCTION

PROXY re-encryption (PRE) [1] provides a secure and flexible method for a sender to store and share data. A user may encrypt his file with his own public key and then store the ciphertext in an honest-but-curious server. When the receiver is decided, the sender can delegate a re-encryption key associated with the receiver to the server as a proxy. Then the proxy re-encrypts the initial ciphertext to the intended receiver. Finally, the receiver can decrypt the resulting ciphertext with her private key. The security of PRE usually assures that (1) neither the server/proxy nor non-intended receivers can learn any useful information about the (re-)encrypted file, and (2) before receiving the re-encryption key, the proxy can not re-encrypt the initial ciphertext in a meaningful way. Efforts have been made to equip PRE with versatile capabilities. The early PRE was proposed in the traditional public- key infrastructure setting which incurs complicated certificate management [2]. To relieve from this problem, several identity-based PRE (IPRE) schemes [3], [4], [5], [6], [7], [8] were proposed so that the receivers' recognizable identities can serve as public keys. Instead of fetching and verifying the receivers' certificates, the sender and the proxy just need to know the receivers' identities, which is more convenient in practice. PRE and IPRE allows a single receiver. If there are more receivers, the system needs to invoke PRE or IPRE multiple times.

To address this issue, the concept of broadcast PRE (BPRE) has been proposed [9]. BPRE works in a similar way as PRE and IPRE but more versatile. In contrast, BPRE allows a sender to generate an initial ciphertext to a receiver set, instead of a single receiver. Further, the sender can delegate a re-encryption key associated with another receiver set so that the proxy can re-encrypt to. The above PRE schemes only allows the re-encryption procedure is executed in an all-or-nothing manner. The proxy can either re-encrypt all the initial ciphertexts or none of them. This coarse-gained control over ciphertexts to be re-encrypted may limit the application of PRE systems. To fill this gap, a refined concept referred to as conditional PRE (CPRE) has been proposed. In CPRE schemes [6], [7], [8], [9], [10], [11], [12], [13], a sender can enforce fine-grained re-encryption control over his initial ciphertexts. The sender achieves this goal by associating a condition with a reencryption key. Only the ciphertexts meeting the specified condition can be re-encrypted by the proxy holding the corresponding re-encryption key. A recent conditional proxy broadcast re-encryption scheme [14] allows the senders to control the time to reencrypt their initial ciphertexts. When a sender generates a re-encryption key to re-encrypt an initial ciphertext, the sender needs to take the original receivers' identities of the initial ciphertext as input. In practice, it means that the sender must locally remember the receivers' identities of all initial ciphertexts. This requirement makes this scheme constrained for the memory-limited or mobile senders and efficient only for special applications.

II. DEFINING CIBPRE AND ITS SECURITY

A CIBPRE system consists of algorithms SetupPRE, ExtractPRE, EncPRE, RKExtractPRE, ReEncPRE, Dec-1PRE and Dec-2PRE. In a typical implementation scenario of a CIBPRE system, a KGA runs algorithms SetupPRE and ExtractPRE respectively to set up a CIBPRE scheme and generate users' private keys according to their identities. A sender runs algorithm EncPRE to encrypt a plaintext, and

generate an initial CIBPRE ciphertext which can be decrypted by some intended receivers, and uploads the ciphertext to a server. Let S be the set of these intended receivers. When a receiver in set S is online, he retrieves the initial CIBPRE ciphertext from the server and runs algorithm Dec-1PRE to decrypt out the plaintext. when a receiver in set S wants to share the plaintext to several new receivers (who are not in set S), he runs algorithm RKExtractPRE to generate a reencryption key and delegate this key to a proxy. Let S0 be the set of these new receivers. The proxy runs algorithm ReEncPRE to re-encrypt the initial CIBPRE ciphertext and generate a re-encrypted CIBPRE ciphertext. When a receiver in set S0 is online, he retrieves the re-encrypted CIBPRE ciphertext from the proxy and runs algorithm Dec-2PRE to decrypt out the plaintext. The formal definition of CIBPRE is as follows.

Definition (CIBPRE). Let N 2 N be the maximal size of receiver set for one CIBPRE encryption or re-encryption. A CIBPRE scheme consists of following algorithms:

- $\mathbf{Setup}_{\mathrm{PRE}}(\lambda, N)$: *Given a security parameter* $\lambda \in \mathbb{N}$ *and value* N*, this algorithm outputs a master public key* $\mathbf{PK}_{\mathrm{PRE}}$ *and a master secret key* $\mathbf{MK}_{\mathrm{PRE}}$.
- $\mathbf{Extract}_{\mathrm{PRE}}(\mathbf{MK}_{\mathrm{PRE}}, ID)$: *Given* $\mathbf{MK}_{\mathrm{PRE}}$ *and an identity* ID*, this algorithm outputs the private key* SK^{ID}_{PRE}.
- $\mathbf{Enc}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, \mathcal{S}, m, \alpha)$: *Given* $\mathbf{PK}_{\mathrm{PRE}}$*, a set* $\mathcal{S}$ *of some identities (where* $|\mathcal{S}| \leq N$*), a plaintext* m *and a condition* α*, this algorithm outputs an initial CIBPRE ciphertext* C.
- $\mathbf{RKExtract}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, ID, SK^{ID}_{\mathrm{PRE}}, \mathcal{S}', \alpha)$: *Given* $\mathbf{PK}_{\mathrm{PRE}}$*, an identity* ID *and its private key* SK^{ID}_{PRE}*, a set* $\mathcal{S}'$ *of some identities (where* $|\mathcal{S}'| \leq N$*) and a condition* α*, this algorithm outputs a re-encryption key* $d_{ID \to \mathcal{S}'|\alpha}$.
- $\mathbf{ReEnc}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, d_{ID \to \mathcal{S}'|\alpha}, C, \mathcal{S})$: *Given* $\mathbf{PK}_{\mathrm{PRE}}$*, a re-encryption key* $d_{ID \to \mathcal{S}'|\alpha}$*, an initial CIBPRE ciphertext* C *and a set* $\mathcal{S}$ *of some identities (where* $|\mathcal{S}| \leq N$*), this algorithm outputs a re-encrypted CIBPRE ciphertext* $\tilde{C}$.
- $\mathbf{Dec\text{-}1}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, ID, SK^{ID}_{\mathrm{PRE}}, C, \mathcal{S})$: *Given* $\mathbf{PK}_{\mathrm{PRE}}$*, an identity* ID *and its private key* SK^{ID}_{PRE}*, an initial CIBPRE ciphertext* C*, and a set* $\mathcal{S}$ *of some identities (where* $|\mathcal{S}| \leq N$*), this algorithm outputs a plaintext.*
- $\mathbf{Dec\text{-}2}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, ID, SK^{ID}_{\mathrm{PRE}}, \tilde{C}, \mathcal{S}')$: *Given* $\mathbf{PK}_{\mathrm{PRE}}$*, an identity* ID *and its private key* SK^{ID}_{PRE}*, a re-encrypted CIBPRE ciphertext* $\tilde{C}$ *and a set* $\mathcal{S}'$ *of some identities (where* $|\mathcal{S}'| \leq N$*), this algorithm outputs a plaintext.*

As usual, a CIBPRE scheme must satisfy the following consistencies:

- *For any initial CIBPRE ciphertext* $C \leftarrow \mathbf{Enc}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, \mathcal{S}, m, \alpha)$ *and any private key* $SK^{ID}_{\mathrm{PRE}} \leftarrow \mathbf{Extract}_{\mathrm{PRE}}(\mathbf{MK}_{\mathrm{PRE}}, ID)$*, if* $ID \in \mathcal{S}$*, then algorithm* $\mathbf{Dec\text{-}1}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, ID, SK^{ID}_{\mathrm{PRE}}, C, \mathcal{S})$ *always outputs the plaintext* m.
- *For any re-encrypted CIBPRE ciphertext* $\tilde{C} \leftarrow \mathbf{ReEnc}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, d_{ID \to \mathcal{S}'|\alpha'}, C, \mathcal{S})$ *and any private key* $SK^{ID'}_{\mathrm{PRE}} \leftarrow \mathbf{Extract}_{\mathrm{PRE}}(\mathbf{MK}_{\mathrm{PRE}}, ID')$*, where* $d_{ID \to \mathcal{S}'|\alpha'} \leftarrow \mathbf{RKExtract}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, SK^{ID}_{\mathrm{PRE}}, \mathcal{S}', \alpha')$*,* $C \leftarrow \mathbf{Enc}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, \mathcal{S}, m, \alpha)$ *and* $SK^{ID}_{\mathrm{PRE}} \leftarrow \mathbf{Extract}_{\mathrm{PRE}}(\mathbf{MK}_{\mathrm{PRE}}, ID)$*, if* $ID \in \mathcal{S} \bigwedge \alpha = \alpha' \bigwedge ID' \in \mathcal{S}'$*, then algorithm* $\mathbf{Dec\text{-}2}_{\mathrm{PRE}}(\mathbf{PK}_{\mathrm{PRE}}, ID', SK^{ID'}_{\mathrm{PRE}}, eC, \mathcal{S}')$ *always outputs the plaintext* m.

A. Our Contribution

In this paper, we refine PRE by incorporating the advantages of IPRE, CPRE and BPRE for more flexible applications and propose a new concept of Proxy Re-encryption Schemes for Data Security in Cloud is a Conditional Identity-based Broadcast PRE (CIBPRE). In a CIBPRE system, a trusted Key Generation Center (KGC) initializes the system parameters of CIBPRE, and generates private keys for users. To securely share files to multiple receivers, a sender can encrypt the files with the receivers' identities and file-sharing conditions. If later the sender would also like to share some files associated with the same condition with other receivers, the sender can delegate a re-encryption key labeled with the condition to the proxy, and the parameters to generate the re-encryption key is independent of the original receivers of these files. Then the proxy can reencrypt the initial ciphertexts matching the condition to the resulting receiver set. With CIBPRE, in addition to the initial authorized receivers who can access the file by decrypting the initial ciphertext with their private keys, the newly authorized receivers can also access the file by decrypting the re-encrypted ciphertext with their private keys.

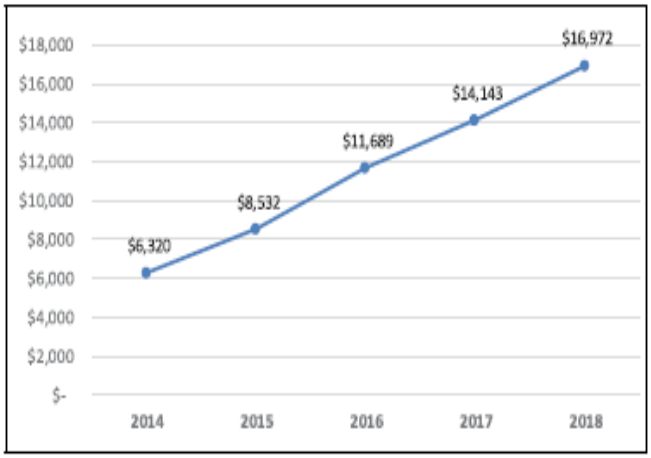

Fig.1. The worldwide revenue forecast for cloud Business Email (unit: Million).

B. Technology detail

Proxy re-encryption schemes are cryptosystems which allow third parties (proxies) to alter a ciphertext which has been encrypted for one party, so that it may be decrypted by another. Proxy re-encryption schemes are similar to traditional symmetric or asymmetric encryption schemes, with the addition of two functions:

Delegation–allows a message recipient (keyholder) to generate a re-encryption key based on his secret key and the key of the delegated user. This re-encryption key is used by the proxy as input to the re-encryption function, which is executed by the proxy to translate ciphertexts to the delegated user's key. Asymmetric proxy re-encryption schemes come in bi-directional and uni-directional varieties. In a bi-directional scheme, the re-encryption scheme is reversible—that is, the re-encryption key can be used to translate messages from Bob to Charlie, as well as from

Charlie to Bob.This can have various security consequences, depending on the application. One notable characteristic of bi-directional schemes is that both the delegator and delegated party (e.g., Charlie and Bob) must combine their secret keys to produce the re-encryption key. A uni-directional scheme is effectively one-way; messages can be re-encrypted from Bob to Charlie, but not the reverse. Uni-directional schemes can be constructed such that the delegated party need not reveal its secret key. For example, Bob could delegate to Charlie by combining his secret key with Charlie's public key.

Transitivity – Transitive proxy re-encryption schemes allow for a ciphertext to be re-encrypted an unlimited number of times. For example, a ciphertext might be re-encrypted from Bob to Charlie, and then again from Charlie to David and so on. Non-transitive schemes allow for only one (or a limited number) of re-encryptions on a given ciphertext. Currently, there is no known uni-directional, transitive proxy re-encryption scheme. It is an open problem as to whether such constructions are possible.A proxy re-encryption is generally used when one party, say Bob, wants to reveal the contents of messages sent to him and encrypted with his public key to a third party, Chris, without revealing his private key to Chris. Bob does not want the proxy to be able to read the contents of his messages. [1] Bob could designate a proxy to re-encrypt one of his messages that is to be sent to Chris. This generates a new key that Chris can use to decrypt the message. Now if Alice sends Chris a message that was encrypted under Bob's key, the proxy will alter the message, allowing Chris to decrypt it. This method allows for a number of applications such as e-mail forwarding, law-enforcement monitoring, and content distribution. Skycryptor [2] uses proxy re-encryption methods for enabling end-to-end encrypted file sharing over modern cloud collaboration applications.

A weaker re-encryption scheme is one in which the proxy possesses both parties' keys simultaneously. One key decrypts a plaintext, while the other encrypts it. Since the goal of many proxy re-encryption schemes is to avoid revealing either of the keys or the underlying plaintext to the proxy, this method is not ideal. Identity-based conditional proxy re-encryption (IBCPRE) is a type of proxy re-encryption (PRE) scheme in the identity-based public key cryptographic setting. An IBCPRE scheme is a natural extension of proxy re-encryption on two aspects. The first aspect is to extend the proxy re-encryption notion to the identity-based public key cryptographic setting. The second aspect is to extend the feature set of proxy re-encryption to support conditional proxy re-encryption. By conditional proxy re-encryption, a proxy can use an IBCPRE scheme to re-encrypt a ciphertext but the ciphertext would only be well-formed for decryption if a condition applied onto the ciphertext together with the re-encryption key is satisfied. This allows fine-grained proxy re-encryption and can be useful for applications such as secure sharing over encrypted cloud data storage.

III. EXISTING SYSTEM

A public-key encryption scheme allows anyone who has the public key of a receiver to encrypt messages to the receiver using the public key in such a way that only the corresponding private key known only to the receiver can decrypt and recover the messages. The public key of a user, therefore, can be published for allowing everyone to use it for encrypting messages to the user while the private key of the user has to be kept secret for the decryption purpose. Both the public key and the corresponding private key of the user are generated by the user in general. Under the identity-based cryptographic setting, the public key of the user can be an arbitrary string of bits provided that the string can uniquely identify the user in the system. The unique string, for example, can be an email address, a phone number, and a staff ID (if used only internally within an organization). However, the corresponding private key is no longer generated by the user. From the public key, which is a unique binary string, there is a key generation center (KGC), which generates and issues the private key to the user. The KGC has a public key, which is assumed to be publicly known, and the encryption and decryption then work under the unique binary string defined public key and the corresponding private key, respectively, with respect to the KGC's public key. Proxy Re-encryption allows a ciphertext, which originally can only be decrypted by a user, to transformed by a public entity, called proxy, to another ciphertext so that another user can also decrypt. Suppose the two users are Alice and Bob.

Alice has some messages: M1, M2,… Mn. She intends to encrypt them under her public key, and then upload the encrypted messages to some server. Now when Alice wants to share these n encrypted messages with Bob, Alice can use a proxy re-encryption scheme to allow the server to re-encrypt these n encrypted messages so that Bob can decrypt these re-encrypted messages directly using his own private key. To do so in the proxy re-encryption scheme, Alice uses her private key and the public key of Bob to generate a re-encryption key. Alice then sends the re-encryption key to the server. Upon receiving this re-encryption key, the server uses the key to transform all the n encrypted messages C1, C2, …, Cn to a new form denoted as D1, D2, …, Dn. Bob can then download D1, D2, …, Dn, decrypt them, and recover the messages M1, M2, … Mn using his private key.
In an identity-based conditional proxy re-encryption (IBCPRE) system, users set their public keys as unique identities of the users. One of the main advantages of using identity-based cryptographic algorithms is the elimination of public key certificates which can help enhance the usability of the target security applications. The term 'Conditional' in IBCPRE refers to an additional feature, which allows each encrypted message to have a 'tag' associated with. In addition to the tag, each re-encryption key also has a 'tag' attached. The IBCPRE is designed so that only if the tag of an encrypted message matches with the tag of a re-encryption key can the encrypted message be re-encrypted.

IV. PROPOSED SYSTEM

In this work, we refine PRE by incorporating the advantages of IPRE, CPRE and BPRE for more flexible applications and propose a new concept of Proxy Re-encryption Schemes for Data Security in Cloud is a Conditional Identity-based Broadcast PRE (CIBPRE). In a CIBPRE system, a trusted Key Generation Centre (KGC) initializes the system parameters of CIBPRE, and generates private keys for users. To securely share files to multiple receivers, a sender can encrypt the files with the receivers' identities and file-sharing conditions. If later the sender would also like to share some files associated with the same condition with other receivers, the sender can delegate a re-encryption key labelled with the condition to the proxy, and the parameters to generate the re-encryption key is independent of the original receivers of these files. Then the proxy can re-encrypt the initial ciphertexts matching the condition to the resulting receiver set. With CIBPRE, in addition to the initial authorized receivers who can access the file by decrypting the initial ciphertext with their private keys, the newly authorized receivers can also access the file by decrypting the re-encrypted ciphertext with their private keys. Note that the initial ciphertexts may be stored remotely while keeping secret. The sender does not need to download and re-encrypt repetitively, but delegates a single key matching condition to the proxy. We propose an efficient CIBPRE that is provably secure in the above adversary model.

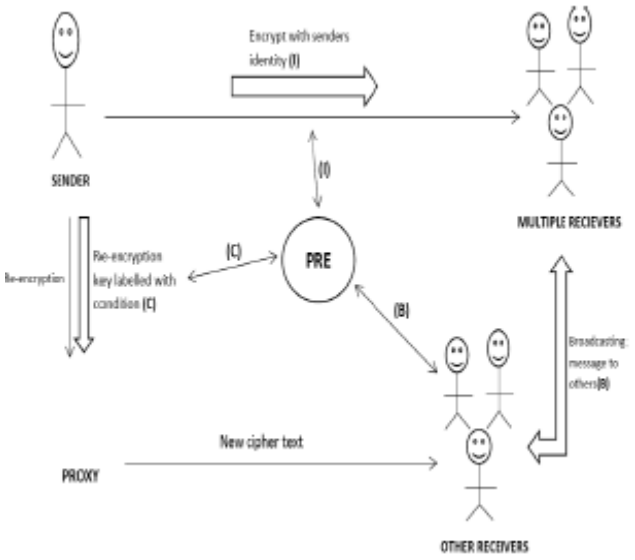

Fig 2: Proposed System architecture For PRE.

We prove that the IND-sID-CPA security of the proposed CIBPRE scheme if the underlying identity-based broadcast encryption (IBBE) scheme is secure and the Decisional Bilinear Diffie-Hellman (DBDH) assumption holds. Our proposed CIBPRE scheme enjoys constant-size initial and re-encrypted ciphertexts, and eliminates the constraints of the recent work in [14]. In a proxy re-encryption (PRE) scheme, suppose Alice gives special information to a proxy that allows it to transform messages encrypted under Alice's public key into an encryption under Bob's public key such that the message is not revealed to the proxy which shown in fig 2 In C-PRE, the proxy also needs to have the right condition key to transform the ciphertext (associated with a condition set by Alice) under Alice's public key into ciphertext under Bob's public key, so that Bob can decrypt it. Here in CIBPRE trusted key generation center handles all the private keys of users. Sender encrypts the plain text with this private key. When sender re-encrypts the cipher text the parameters generate the re-encryption key one independent of original receiver. While converting plain text to cipher text the data is encrypted with the senders identity and file sharing conditions. This cipher text is then sends to the multiple receivers as shown in fig 2. These multiple receivers then broadcast the cipher text to other multiple receivers. These condition, identity and broadcasting of cipher text forms CBI Proxy Re-Encryption. Initial ciphertext can be stored remotely and secretly on the proxy server so that senders don't have to download the plain text all the time and encrypt it repetitively. Only delegating condition repetitively to multiple users is important and condition is based on the single key matching condition. Ciphertext cannot be re-encrypted correctly to new ciphertext if key and ciphertext are having different conditions.

V. METHODOLOGY

The CIBPRE-based cloud email system consists of a trusted KGC (built by an enterprise administrator), a cloud server and users. The CIBPRE-based cloud email system works as follows:

Initialization: In this phase, the KGC generates the system parameters to initialize the CIBPRE-based cloud email system. It chooses a security parameter $\lambda \varepsilon N$ and a value N N (the maximal number of receivers of an email), and runs algorithm SetupPRE(λ,N) to generate a pair of master public and secret keys PKPRE and MKPRE. It chooses a secure symmetric key encryption scheme, i.e. AES (the popular choice in practice). Without loss of generality, let the chosen symmetric key encryption scheme be (X; SEx; SDx), where X GT PKPREis the symmetric key space, SEx and SDxrespectively denotes the encryption and decryption gorithms both with a symmetric key x X. Finally, it publishes (PKPRE;X; SEx; SDx).

Key Management: In this phase, when a new user joins this system, the KGC generates a private key for him. Without loss of generality, let ID denote the email address of the new user. The KGC runs algorithm ExtractPRE(MKPRE; ID) to generate the private key SKIDPRE , and sends it to the user in a secure channel which is established by the SSL/TLS protocol.

Send An Encrypted Cloud Email: In this phase, a user can send an encrypted email to other users. And this email will be stored in the cloud server. If the user wants to review this email, he can fetch the encrypted email from the cloud server and decrypt it. Suppose user ID1 wants to send the email content F (including the associated attachment) to the users {ID'2 …ID'n} (where n <= N).

Forward A History Encrypted Cloud Email: In this phase, a user can forward a history encrypted email to new users by generating a re-encryption key for these users and the subject of this email.

Performance: In the above steps, the capability "Identity-based" of CIBPRE avoid user ID1 to fetch and verify the certificates of users {ID'2 …ID'0n } before generating a re-encryption key. The capability "Broadcast" of CIBPRE makes the generated re-encryption key having the constant size.

Initialization: In this phase, the KGC generates the system parameters to initialize the CIBPRE-based cloud email system. It chooses a security parameter and runs algorithm generate a pair of master public and secret keys. It chooses a secure symmetric key encryption scheme, i.e. AES.

Key Management: In this phase, when a new user joins this system, the KGC generates a private key for him.

Send An Encrypted Cloud Email: In this phase, a user can send an encrypted email to other users. And this email will be stored in the cloud server. If the user wants to review this email, he can fetch the encrypted email from the cloud server and decrypt it.

Forward A History Encrypted Cloud Email: In this phase, a user can forward a history encrypted email to new users by generating a re-encryption key for these users and the subject of this email.

VI. APPLICATIONS

A. Cloud Email System: A Promising Application

Cloud email system allows an enterprise to rent the cloud SaaS service to build an email system. It is much cheaper and scalable than traditional on-premises solution. In 2014, the Radiated Group [17] showed the worldwide revenue forecast for Cloud Business Email, from 2014 to 2018. The Cloud Business Email market is expected to generate. In 2012, the Proof point Group [18] used an economic model that estimates opportunities for quantifiable cost savings of cloud email system compared with traditional on-premises email system. The Proof point model calculates expenses for both systems at the time of acquisition as well as over a four-year period, such as software licensing costs, hardware and storage costs, service expenses, operational expenses. Table 1 summarizes savings using the economic model. Note that NAS (Network Attached Storage) and CAS (Content-addressable storage) in this table are two different technologies which are usually applied in many storage systems. Cloud email system is a promising and important application due to its advantageous features. We build an encrypted cloud email system with CIBPRE. It allows a user to send an encrypted email to multiple receivers, store his encrypted emails in an email server, review his history encrypted emails, forward his history encrypted emails of the expected subject to multiple new eceivers. Moreover, the cost of an extra email header to achieve this goal is the constant.

Compared with existing approaches such as Privacy Good Privacy (PGP) protocol [15] and Identity-Based Encryption (IBE) [16], our CIBPRE-based system is implementation-friendly and more efficient in communication. In PGP, a sender first verifies a receiver's certificate and encrypts an email by the receiver's public key; then the receiver decrypts the received email with his private key. IBE avoids the certificate verification of PGP. Using IBE, a sender directly encrypts an email using a receiver's email address. Though both PGP and IBE keep the security of cloud email, their performances are less than CIBPRE. When a sender wants to send an encrypted email to multiple receivers, the size of the ciphertext generated by CIBPRE is constant. In contrast, both PGP and IBE cause the size linear with the number of receivers. When a sender wants to forward a historically encrypted email to multiple receivers, CIBPRE only requires the sender to generate a re-encryption key (with constant size) and send the key to cloud, and then the cloud re encrypts the email and generates a constant size ciphertext for these receivers. In contrast, with PGP or IBE, the sender must fetch the historically encrypted email from the cloud and decrypt it, and then re-encrypt it again to these receivers one by one. Therefore, CIBPRE is very suitable for building encrypted cloud email systems and our proposed CIBPRE scheme is more convenient than PGP and IBE to keep the security of cloud email system.

B. Security Analysis

In a proxy re-encryption scheme a semi-trusted proxy converts a ciphertext for Alice into a ciphertext for Bob without seeing the underlying plaintext. A number of solutions have been proposed in the public-key setting. In this paper, we address the problem of Identity-Based proxy re-encryption, where ciphertexts are transformed from one identity to another. Our schemes are compatible with current IBE deployments and do not require any extra work from the IBE trusted-party key generator. In addition, they are non-interactive and one of them permits multiple re-encryptions. Their security is based on a standard assumption (DBDH) in the random oracle model. Recently, a number of extended Proxy Re-Encryptions (PRE), e.g. Conditional (CPRE), identity-based PRE (IPRE) and broadcast PRE (BPRE), have been proposed for flexible applications. By incorporating CPRE, IPRE and BPRE, this report proposes a versatile primitive referred to as Proxy Re-encryption Schemes for Data Security in Cloud is a Conditional Identity-based Broadcast PRE (CIBPRE) and formalizes its semantic security. CIBPRE allows a sender to encrypt a message to multiple receivers by specifying these receivers' identities, and the sender can delegate a re-encryption key to a proxy so that he can convert the initial ciphertext into a new one to a new set of intended receivers. Moreover, the re-encryption key can be associated with a condition such that only the matching ciphertexts can be re-encrypted, which allows the original sender to enforce access control over his remote ciphertexts in a fine-grained manner. We propose an efficient CIBPRE scheme with provable security.

In the instantiated scheme, the initial ciphertext, the re-encrypted ciphertext and the re-encryption key are all in constant size, and the parameters to generate a re-encryption key are independent of the original receivers of any initial ciphertext. Finally, an application of CIBPRE to secure cloud email system advantageous over existing secure email systems based on Pretty Good Privacy protocol or identity-based encryption. The first consistency is straightforward. It means that any initial CIBPRE ciphertext can be correctly decrypted by its intended receivers. The second consistency is a somewhat sophisticated. Its main idea is to define that any correctly reencrypted CIBPRE ciphertext can be correctly decrypted by its intended receivers. Therefore, to define the second consistency, we must define what is a correctly re-encrypted CIBPRE ciphertext, and define who can correctly decrypt the ciphertext. For any re-encrypted CIBPRE ciphertext of an initial CIBPRE ciphertext by a re-encryption key, the second consistency defines that the re-encrypted CIBPRE ciphertext is correct, if the generator of the re-encryption key is an intended receiver of the initial CIBPRE ciphertext, and the initial CIBPRE ciphertext and the re-encryption key has the same condition. Also it defines that the correctly reencrypted CIBPRE ciphertext can be correctly decrypted by the receivers who are defined by the re-encryption key.

We next define the IND-sID-CPA security of CIBPRE. Roughly speaking, the security means that no PPT adversary can decide which one of two plaintexts is encrypted by an initial CIBPRE ciphertext, if he does not know the private keys of the intended receivers both of the initial CIBPRE ciphertext and its re-encrypted CIBPRE ciphertexts. In other words, without the corresponding private keys, an initial CIBPRE ciphertext and its re-encrypted CIBPRE ciphertexts leak nothing about their encrypted plaintext. The IND-sID-CPA security of CIBPRE defines an attack game between a PPT adversary and a challenger. The attack game consists of several phases. In the initialization phase, the adversary commits a set S_ of challenge identities and a challenge condition a_ that he wants to attack. In the setup phase, the challenger sets up a CIBPRE scheme. In the challenge phase, the adversary chooses two challenge plaintexts; the challenger randomly chooses one of these two plaintexts, encrypts the chosen plaintext by the committed S* and a* to generate an initial CIBPRE ciphertext (it also is a challenge ciphertext in the attack game), and asks the adversary to decide which one of these two plaintexts is encrypted. Before and after the challenge phase, the adversary is allowed to query some identities' private keys and re-encryption keys. It means that the adversary can collude with some users in practice. But the adversary cannot query the private keys of the challenge identities in set S* and the private keys which can decrypt the re-encrypted CIBPRE ciphertext of the challenge ciphertext. If the adversary has no advantage to make a correct decision, then we say that the CIBPRE scheme is IND-sID-CPA secure.

VII. CONCLUSION

We proposed and Efficient proxy re-encryption scheme for secure data sharing. It allows a user to share the encrypted data with other users. All users take their identities as public keys to encrypt data. It avoids a user to fetch and verify other users' certificates before encrypting his data. Moreover, it allows a user to generate a broadcast ciphertext for multiple receivers and share his outsourced encrypted data to multiple receivers in a batch manner.

VIII. REFERENCES

[1] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in Proc. Int. Conf. Theory Appl. Crytographic Techn.: Adv. Cryptol., 1998, pp. 127–144.

[2] A. Boldyreva, M. Fischlin, A. Palacio, and B. Warinschi, "A closer look at PKI: Security and efficiency," in Proc. 10th Int. Conf. Practice Theory Public-Key Cryptography, 2007, pp. 458–475.

[3] M. Green and G. Ateniese, "Identity-based proxy re-encryption," in Proc. 5th Int. Conf. Appl. Cryptography Netw. Security, 2007, pp. 288–306.

[4] T. Matsuo, "Proxy re-encryption systems for identity-based encryption," in Proc. 1st Int. Conf. Pairing-Based Cryptography, 2007, pp. 247–267.

[5] C.-K. Chu and W.-G. Tzeng, "Identity-based proxy re-encryption without random oracles," in Proc. 10th Int. Conf. Inf. Security, 2007, pp. 189–202.

[6] L. Ibraimi, Q. Tang, P. Hartel, and W. Jonker, "A type-and-identity-based proxy re-encryption scheme and its application in healthcare," in Proc. 5th VLDB Conf. Secure Data Manage., 2008, pp. 185–198.

[7] J. Shao, G. Wei, Y. Ling, and M. Xie, "Identity-based conditional proxy re-encryption," in Proc. IEEE Int. Conf. Commun., 2011, pp. 1–5.

[8] K. Liang, Z. Liu, X. Tan, D. S. Wong, and C. Tang, "A CCA-secure identity-based conditional proxy re-encryption without random oracles," in Proc. 15th Int. Conf. Inf. Security Cryptol., 2012, pp. 231–146.

[9] C.-K. Chu, J. Weng, S. S. M. Chow, J. Zhou, and R. H. Deng, "Conditional proxy broadcast re-encryption," in Proc. 14th Australasian Conf. Inf. Security Privacy, 2009, pp. 327–342.

[10] Q. Tang, "Type-based proxy re-encryption and its construction," in Proc. 9th Int. Conf. Cryptol. India: Progress Cryptol., 2008 pp. 130–144.

[11] J. Weng, R. H. Deng, X. Ding, C.-K. Chu, and J. Lai, "Conditional proxy re-encryption secure against chosen-ciphertext attack," in Proc. 4th Int. Symp. Inf., Comput. Commun.Security, 2009, pp. 322–332.

78 IEEE TRANSACTIONS ON COMPUTERS, VOL. 65, NO. 1, JANUARY 2016

[12] J. Weng, Y. Yang, Q. Tang, R. H. Deng, and F. Bao, "Efficient conditional proxy re-encryption with chosen-ciphertext security," in Proc. 12th Int. Conf. Inf. Security, 2009, pp. 151–166.

[13] L. Fang, W. Susilo, and J. Wang, "Anonymous conditional proxy re-encryption without random oracle," in Proc. 3rd Int. Conf. Provable Security, 2009, pp. 47–60.

[14] K. Liang, Q. Huang, R. Schlegel, D. S. Wong, and C. Tang, "A conditional proxy broadcast re-encryption scheme supporting timedrelease," in Proc. 9th Int. Conf. Inf. Security Practice Experience, 2013, pp. 132–146.
[15] P. R. Zimmermann, PGP Source Code and Internals. Cambridge, MA, USA: MIT Press, 1995.
[16] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in Proc. 21st Annu. Int. Cryptol.: Adv. Cryptol., 2001, pp. 213–239.
[17] Radicati Group. (2014). Cloud business email market, 2014-2018 [Online]. Available: http://www.radicati.com/ wp/wp-content/uploads/2014/10/Cloud-Business-Email-Market-2014-2018-Executive-Summary.pdf
[18] Proofpoint Group. (2012). Cloud-based archiving vs. on-premises legacy archiving [Online]. Available: http://video.proofpoint. com/id/cloud-based-archiving-vs.-on-premises-legacy-archiving-TCO-white-paper
[19] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy reencryption schemes with applications to secure distributed storage," ACM Trans. Inf. Syst. Security, vol. 9, pp. 1–30, 2006.
[20] R. H. Deng, J. Weng, S. Liu, and K. Chen, "Chosen-ciphertext secure proxy re-encryption without pairings," Cryptol. Netw. Security, vol. 5339, pp. 1–17, 2008.
[21] V. Kirtane and C. P. Rangan, "RSA-TBOS signcryption with proxy re-encryption," in Proc. 8th ACM Workshop Digital Rights Manage., 2008, pp. 59–66.
[22] B. Libert and D. Vergnaud, "Unidirectional chosen-ciphertext secure proxy re-encryption," in Proc. 11th Int. Workshop Practice Theory, 2008, pp. 360–379.
[23] J. Shao and Z. Cao, "CCA-secure proxy re-encryption without pairings," in Proc. 12th Int. Conf. Practice Theory Public Key Cryptography,2009, pp. 357–176.
[24] G. Ateniese, K. Benson, and S. Hohenberger, "Key-private proxy re-encryption," in Proc. Cryptographers' Track RSA Conf. Topics Cryptol., 2009, pp. 279–294.
[25] J. Shao, P. Liu, G. Wei, and Y. Ling, "Anonymous proxy reencryption," Security Commun. Netw., vol. 5, no. 5, 2012, pp. 439–449.
[26] R. Canetti and S. Hohenberger, "Chosen-ciphertext secure proxy reencryption," in Proc. 14th ACM Conf. Comput. Commun. Security, 2007, pp. 185–194.
[27] T. Matsuda, R. Nishimaki, and K. Tanaka, "CCA proxy re-encryption without bilinear maps in the standard model," in Proc. 13th Int. Conf. Practice Theory Public Key Cryptography, 2010, pp. 261–278.
[28] K. Liang, M. H. Au, J. K. Liu, X. Qi, W. Susilo, X. P. Tran, D. S. Wong, and G. Yang, "A DFA-based functional proxy re-encryption scheme for secure public cloud data sharing," IEEE Trans. Inf. Forensics Security, vol. 9, no. 10, pp. 1667–1680, Oct. 2014.
[29] K. Liang, J. K. Liu, D. S. Wong, and W. Susilo, " An efficient cloudbased revocable identity-based proxy re-encryption scheme for public clouds data sharing," in Proc. Eur. Symp. Res. Comput. Security, 2014, pp. 257–272.
[30] C. Delerabl_ee, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in Proc. 13th Int. Conf. Theory Appl. Cryptol. Inf. Security: Adv. Cryptol., 2007, pp. 200–215.
[31] D. Boneh and X. Boyen, "Efficient selective-id secure identitybased encryption without random oracles," in Proc. Adv. Cryptol., 2004, pp. 223–238.
[32] D. Boneh and X. Boyen, "Secure identity based encryption without random oracles," in Proc. 24th Annu. Int. Cryptol. Conf.: Adv. Cryptol., 2004, pp. 197–206.