PreBAC: a novel Access Control scheme based Proxy Re-Encryption for cloud computing

Mang Su[1*] and Liangchen Wang[2]
[1] School of Computer Science and Engineering
Nanjing University of Science and Technology
Jiangsu, Nanjing, China
[e-mail: sumang@njust.edu.cn]
[2] Nanjing Municipal Public Security Bureau
Jiangsu, Nanjing, China
[e-mail: wlc8571@163.com]
*Corresponding author: Mang Su

Received September 24, 2017; revised November 6, 2018; accepted November 13, 2018; published May 31, 2019

Abstract

Cloud computing is widely used in information spreading and processing, which has provided a easy and quick way for users to access data and retrieve service. Generally, in order to prevent the leakage of the information, the data in cloud is transferred in the encrypted form. As one of the traditional security technologies, access control is an important part for cloud security . However , the current access control schemes are not suitable for cloud , thus, it is a vital problem to design an access control scheme which should take account of complex factors to satisfy the various requirements for cipher text protection. We present a novel access control scheme based on proxy re-encryption(PRE) technology (PreBAC) for cipher text. It will suitable for the protection of data confidently and information privacy. At first, We will give the motivations and related works, and then specify system model for our scheme. Secondly, the algorithms are given and security of our scheme is proved. Finally, the comparisons between other schemes are made to show the advantages of PreBAC.

Keywords: multi-factor access control, proxy re-encryption(PRE), cipher text protection, cloud computing

http://doi.org/10.3837/tiis.2019.05.028 **ISSN : 1976-7277**

1. Introduction

With the developments of related technologyies, cloud computing provides a crucial important support for spreading of information. The users can obtain the service and data by the renting way. More and more information has appeared in cloud including some confidential data and personal privacy. The user expects to benefit from cloud without leaking their information. Thus, it is an hot topic of cloud security to keep the data and privacy from stealing and destroying [1]. However, cloud possesses characteristic traits that set apart and distinguish it from other information systems. First, the cloud needs to process an extreme volume of data and users; secondly, the relationship between the users and data are complicated. Thirdly, there are variable services in cloud, including the software, platform and infrastructure; Finally, the forms of cloud deploying are complicated. All of them bring a serious situation to cloud security, which could be summarized as followed.

1) A plenty of the cloud service providers adopt the "username-password" to realize the access control and attestation, which is not enough for the complex environment. The schemes for data management should take consideration of multi-factor including the identity and role of subject or the temporal and environmental states.

2) The data will be control by cloud service provider instead of its owner. The user's privacy might be dangerous in the plain text form of their data. Therefore, the data in cloud should be encrypted and the schemes for data protection should be designed for the cipher text.

3) There are not enough schemes designed for both multi-factor and cipher text.

To protect the information in cloud, a plenty of the traditional technologies are applied in cloud, such as access control. Access control is mainly used for confidentiality and availability of information and system. Therefore, some researchers have paid attentions to describe the time and location factors, and there appeared a lot of corresponding models and schemes. For instants, Jha et al[2] proposed the Temporal Role-Based Access Control (TRBAC), which describe the time factor based RBAC. It is be beneficial to time-based cloud services. Li at al[5] presented a SecLoc scheme, which offers protection for cloud data following the storage location restrictions. And some works focus on refining the role factors, Yang et al[3] presents the CARBAC model. This model defines the roles in cloud as two parts, including the data generator and users, which makes the roles administration more reasonably. But it requires the data owner to pay more resource to generate and manage the information and roles, which will be a waste of the ability of cloud servers. Some researchers take the other factors to access control in cloud, such as trust [4]. However it is also an important problem to combine the multi-factor and cryptography together. A series of literature have shown some cryptography based access control schemes, for instant, Zhou et al[6] [7] designed JBE. And IBE and ABE are common used in cloud.

However, to realized the schemes in cloud, the data owners are responsible for the information generation and encryption, which will spend a plenty of resource and time. Meanwhile the work above should not be assigned to cloud servers for the information security and privacy. Thus, if we take the multi-factor to cloud access control, there will be a huge cost for the common users who are lack of resource.

Based on the above researches and problems, we will propose a novel access control scheme based on PRE technology (PreBAC). Firstly, we will analyze the motivations for the new scheme, and then describe the system model based the related works. Our scheme will be

shown as tow parts, one is data creation, and the other one is data access. Secondly, the algorithm will be constructed based PRE, and the how to generate the re-encryption key based on the multi-factor of access control will be explain. Thirdly, we will prove the security of the scheme. Finally, the comparisons between PreBAC and other works will be made.

The rest of this paper is organized as follows: The motivations and related work are shown in Section 2 and 3 respectively. The aims and assumptions of PreBAC scheme are described in Section 4, and the system models, main stages and algorithms of it in Section 5. Security proof and properties analysis are discussed in Section 6, the concluding remarks are in Section 7.

2. Motivations

In this section we will show the motivations by an example. It represents a subset of a practical system. The example should be deployed in public cloud. Here we define individual users of the system "Alice","Bob" and "Carl". Among them "Alice" is the data owner, who wants to generate and share the data "*Alice_data*";"Bob" and "Carl" are the users, who will access control the "*Alice_data*", and they are assigned the different access permissions as follows.

For user "Bob", he could access the "*Alice_data*" during 9:00-12:00 in his office.

For user "Carl", she could access the "*Alice_data*" during 15:00-19:00 without the location constrain. "Alice" wants to keep the information from stealing and destroying. Therefore, she submits the "*Alice_data*" in the encrypted form. And to share with other "Bob" and "Carl", she needs to generate different ciphertext of "*Alice_data*" for them. The cloud service provider is honest but curious, who will finish your task for your paid and also be interested in your privacy. So, "Alice" could only finish the work of data generation by herself in the traditional system, which will be a huge work for the complex cloud environment. Our work will focus on how to reduce "Alice's" cost as well as protect the information.

3. State-of-the-art

PRE is a novel cryptographic technology, which proposed from the public-key ideal, according to which a user could encrypt the information with the public key of himself and obtians the cipher text which could be decrypted by his private key. Then he submits this cipher text to the PRE server for re-encryption. The server will finish re-encryption with the re-encryption keys list and get the new cipher text for the other user. Based on PRE, the data owner only needs to generate the original cipher text of the information, and the PRE server will be responsible for re-encryption and sharing instead of the data owner. The server finish its work based on the cipher text, thus it is security and suitable for the data managing for cloud, which also could be useful for personal users to reduce the cost.

However, the PRE cannot be used for cloud independently, and it should be realized with other technologies, such as with IBE and ABE[8][9] .But the descriptions of the attribute and identity are complicated, the certificate is appeared and PRE scheme based on certificate is proposed[10]. During to the time factor of access control, Liu et al. [11] gave a time based PRE scheme. Meanwhile, location, role or other factors of access control are also important for the PRE scheme. Yang et al. [12] proposed the PRE based on conditions. In the previous work, we have proposed a PRE scheme by describing more access conditions [14]. The works above focus on the subject's conditions. For the fine-grained management for objects, Tang et al.[13]

take the ciphertext type as the factor for PRE scheme, but the users should use different keys for different cipher text. It will be a huge and terrible work for common users. Our work will propose PreBAC based on [13][14].

4. Aims and Assumptions

4.1 System aims

The PreBAC we proposed in this paper will base on the works above, and aim at the goals as followed:

1) Complex access control factors description for cipher text: PreBAC will aim to the multiply access control factors description for the cipher text in cloud.

2) Fine-grained data management: PreBAC will satisfy the demand for fine-grained management for ciphertext.

3) User-centered design: PreBAC will decrease the users' computational requirements and reduce the cost for the tenants to use and store the keys.

4) Resistance to attacks: PreBAC will prevent the hackers from attacking the system, for example, brute force, statistical attack and collusion attack.

4.2 System asumptions

PreBAC system is designed and realized under the assumptions as follows.

1) Network connection: All the users can connect to the network freely, and they will pay to the cloud server and submit or access data.

2) Trusted parts in the system: there are three trusted parts, including KGC(Key Generation Centre), data creators and common users, they will not lose the information and their keys actively.

3) Half-trusted parts in the system: there are three trusted parts, including PRE(Proxy re-encryption server), KM(Key management serve), PM(Policy management server) , which are the HBC(honest but curious) systems.

4) Untrusted parts in the system: the cloud data servers are untrusted.

5. PreBAC Scheme

5.1 system model

The notations in PreBAC are shown as follow.

Table 1. notations in PreBAC

notation	description
pk_i	user *i*'s public key
sk_i	user *i*'s private key
$K(M)$	the symmetric cipher text of *M* encrypted by key *k*
$E(K)_i$	the cipher text which could be decrypted by sk_i
$Cert_i$	user *i*'s certification
Con_i	user *i*'s access condition
P_MFAC	access policy
$rk_{i\to j}$	re-encryption key from *i* to *j*

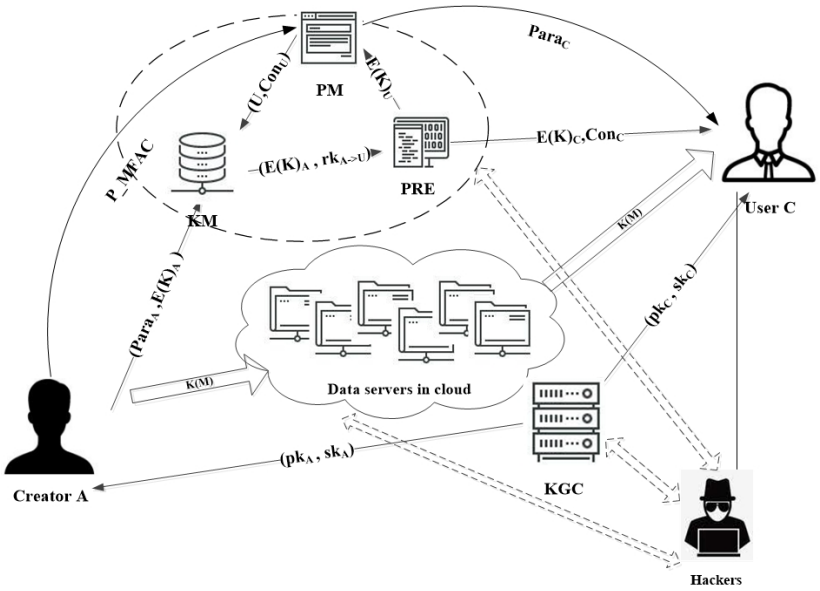


Fig. 1. system model of PreBAC

In order to show the PreBAC scheme, we will show the system model at first (see **Fig. 1.**).

The entities in the system are shown as follows

1)Creator A: She is the data owner, who will generate the data M its ciphertext. And then she will submit the ciphertext to cloud for sharing;

2)User U: He plans to access the data M;

3)Data servers in cloud: Servers of cloud to store the ciphertext of data M;

4)Access control servers: It is not the name for one kind of server. There are three parts: policy managing server(PM), key managing server(KM) and proxy re-encryption server (PRE). They are responsible for data permission management.

5)Hackers: There might be some hackers, who has no permission to access the data in cloud, but they are trying to get it by some network attack means.

5.2 System Stages

PreBAC includes two stages: data uploading and data downloading.

1) Information uploading

This stage is started by data creator *A*. In this stage, A will generate the data M and encrypt it by symmetric algorithm. And then A will encrypt the symmetric key by her public key. Finally, she will submit all the data to cloud server. The detail is shown in **Fig. 2**.

Step1: A uses the symmetric method to generate K(M) by the *k*;

Step2: A submits K(M) to data centre in cloud;

Step3: A encrypts *k* in Step1 by public cryptographic technology based her public key.

Step3-1: A sets the system initiations and key generation instruction to KGC by submitting the parameter *q*.

Step3-2: KGC obtains A's instruction and parameter *q*, and generate the security parameter list $param$ by the method $Setup(q)$.

Step3-3: KGC generates (pk_A, sk_A) by the method $KeyGen(param)$ for A.

Step3-4: A generates the orignal cipher text of *k* and $para_A$ for permission assignment by the method $First\_Enc(K, pk_A)$, and returns $E(K)_A$ and $para_A$to KM.

Step4: A submits the policy for M permission assignment to PM.

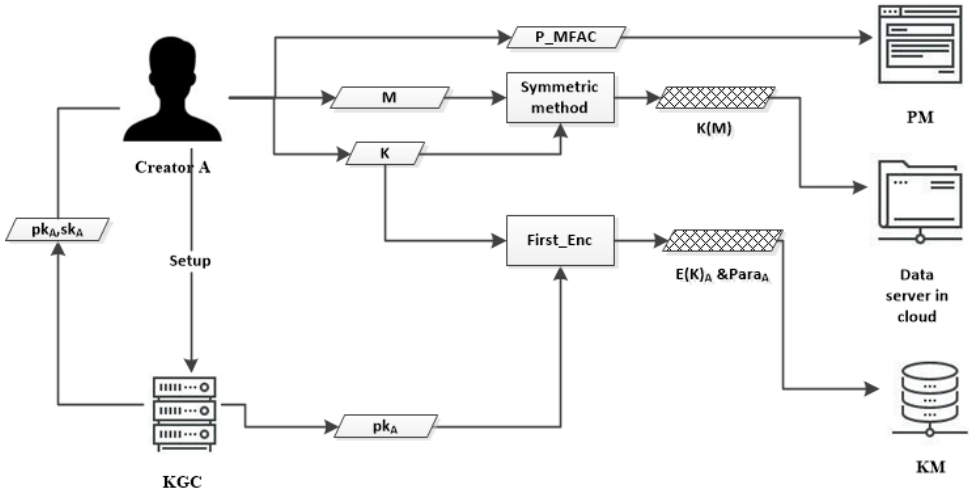


Fig. 2. Information uploading stage

2) Information downloading

This stage is started by data common user *U*. In this stage, *U* will submit the requirement for data accessing and obtain the information he wants. The detail is shown in **Fig. 3**.

Step1: *U* sends the requirement of K(M) to data servers in cloud and download it;

Step2: *U* submits the requirement for $E(K)_U$to KM and download it

Step2-1: *U* gives the certification including his public key to KM;

Step 2-2: *U* submits the requirement for data accessing to PM;

Step2-3: PM gets *U* 's accessing requirement, then collects *U*'s accessing conditions. After that, PM compares the conditions with P_MFAC. If the description in P_MFAC includes the conditions, PM will return the parameter (U, Con_U) to KM.

Step2-4: KM gets the PM's parameter, and searches the A's and U 's certifications respectively, based on which KM generates the re-encryption key and sends to PRE function$ReKeyGen(Para_A, Para_U, Con_U)$.

Step2-5: PRE generates the $E(K)_U$ by function $ReEnc(E(K)_A, rk_{A\to U})$ based on KM's information, and sends it to PM.

Step 2-6: PM gives ($E(K)_U$,Con_U) to *U*.

Step3: *U* obtains M

Step3-1: *U* decrypts the $E(K)_U$ for symmetric key *k* by function $Dec_2(sk_U, E(K)_U, con_U)$.

Step3-2: U decrypts the $K(M)$ *k* to get M.

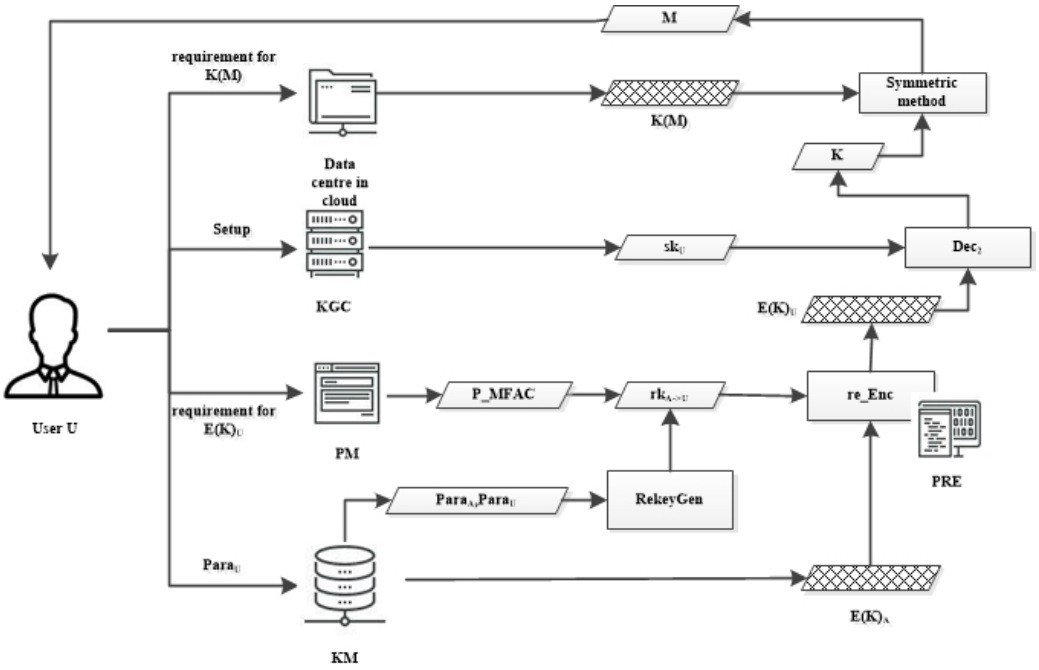


Fig. 3. Information downloading stage

5.3 Description of the access control policy (P_MFAC)

In order to implement the access control based on PRE, the data creators have submitted the access control policy to the PM in the form of P_MFAC. The P_MFAC is defined as a two-tuples (ID_O, $P\text{-}con_U$). ID_O is the ID of the object in cloud, $P\text{-}con_U$ is the constraint to the user who will access the object. ID_O is assigned by the cloud servers. $P\text{-}con_U$ is consisted of several parts, including the user's ID, name, role, temporal state, location or the operation to the objects, such as, "read", "write" or "append".

When an information system will apply the PreBAC, the access conditions would be described based the XML language, and then access conditions could be generated the $P\text{-}con_U$ based on some hash function.

Here is an example of the description of the access condition.In the example, User's name is"Li Lei",he is a student, and he will read the object of No.2 from 2018 Nov. 6th to 7th in the class room.

```
<subject>
        <name>Li Lei<\name>
        <ID>1<\ID>
        <Role>Student<\Role>
        <time>
                <start>20181106<\start>
                <end>20181107<\end>
        <\time>
        <location>classroom<\location>
<\subject>
<operation>Read<\operation>
<Object>
        <ID>2<\ID>
<\Object>
```

5.4 Algorithm

There are seven functions in algorithms of PreBAC.

1) $Setup(q) \rightarrow param$

Input: parameter *q*, define the length of prime *p*.

Output: security parament set $param$, conditions set con_U, and bilinear map e to finish the system parameters initialization

At first, let us pick a q-bit prime p, and defines multiplicative cyclic groups G_1, G_2 of prime order p. And then pick up g as a generator of G_1.

Secondly, we define a hash function list including H_1, H_2, H_3, H_4 with $H_1 : \{0,1\}_* \rightarrow G_1$, $H_2 : \{0,1\}_* \rightarrow Z_p^*$, $H_3 : G_2 \rightarrow \{0,1\}^l$, $H_4 : \{0,1\}^* \rightarrow G_1$.

Finally, $param = \{p, G_1, G_2, g, H_i (i = 1, ..., 4)\}$.

Meanwhile, we define the conditions of access control set $con_U = \{0,1\}*$ and a bilinear map $e : G_1 \times G_1 \rightarrow G_2$.

2) $KeyGen(param) \rightarrow (sk_i, pk_i)$

Input: security parament set $param$.

Output: private and public key for users.

Let us pick $x_i, x_j \in Z_p^*$, and obtain $sk_i = x_i, pk_i = g^{x_i}$.

3) $First\_Enc(k, pk_A) \rightarrow E(k)_A$

Input: plaintxt *K*, public key pk_A.

Output: cipher text of first encryption by creator's public key, parameter of creator A for re-encryption keys.

At first, we pick up $i \in G_2$ to compute $r = H_2(k \parallel i)$, and outputs $E(k)pk_A = (c_1, c_2, c_3, c_4, c_5)$.

$c_1 = g^r$;

$c_2 = k \cdot e(pk_A, H_1(pk_A))^r$;

$c_3 = k \oplus H_3(i)$;

$c_4 = H_1(pk_A)$;

$c_5 = H_4(c_1||c_2||c_3||c_4)^r$.

$para_A = H_1(pk_A)^{sk_A}$

4) $ReKeyGen(para_A, para_U, con_U) \rightarrow rk_{A \rightarrow U}$

Input: condition con_U, $para_A$ and $para_U$.

Output: re-encryption keys.

here $rk_{A \rightarrow U} = (pk_U, pk_U^r, H_1(pk_U \parallel con_U) \cdot para_A, g^{-r})$.

5) $ReEnc(E(k)_A, rk_{A\to U}) \to E(k)_U$

Input: ciphertext of first encrption and re-encryption key.

Output: re-encryption cipher text.

If $e(c_1, H_4(c_1 \parallel c_2 \parallel c_3 \parallel c_4)) = e(g, c_5)$ then $E(k)_U = (c'_1, c'_2, c'_3, c'_4, c'_5)$, otherwise, outputs the error information. The cipher text $E(k)_U$ can be decrypted with sk_U under the corresponding conditions con_U.

$c'_1 = c_1$;

$c'_2 = c_2 \cdot e(pk_U^r g^{-r}, H_1(pk_A)^{-sk_A}) \cdot e(pk_U^r, H_1(pk_U \parallel con_U) \cdot para_A) = i \cdot e(pk_U^r, H_1(pk_U || con_U))$;

$c'_3 = c_3$;

$c'_4 = H_1(pk_U)$;

$c'_5 = H_4(c'_1, c'_2, c'_3, c'_4)^r$;

6) $Dec_1(sk_A, E(k)_A) \to k$

Input: cipher text of first encryption and creator A's private key.

Output: plaintext of *k*.

If $e(c_1, H_4(c_1 \parallel c_2 \parallel c_3 \parallel c_4)) = e(g, c_5)$ goes on, otherwise, it returns error information for integrity.

$i = c_2 / e(c_1, c_4)^{sk_A}$;

$k = c_3 \oplus H_3(i)$;

$r = H_2(k \parallel i)$, if $c_1 = g^r$ and $c_2 = i \cdot e(g, c_4)^{r sk_A}$ then outputs k.

7) $Dec_2(sk_U, E(k)_U, con_U) \to k$,

Input: cipher text of re-encryption and user U's private key with condition *con*$_U$.

Output: plaintext of *k*.

If $e(c'_1, H_4(c'_1 \parallel c'_2 \parallel c'_3 \parallel c'_4)) = e(g, c'_5)$ goes on, otherwise, it returns error information for integrity.

$i = c'_2 / e(c'_1, H_1(pk_j || con_U))^{sk_U}$;

$k = c'_3 \oplus H_3(i)$;

$r = H_2(m \parallel i)$,if $c'_1 = g^r$and $c'_2 = i \cdot e(pk_U, H_1(pk_U \parallel con_U))^r$, then outputs k

6. Discussion

6.1 Properties

PreBAC has combined PRE and access control together. Based on PreBAC, if a data owner wants to share data *M*, she should encrypt *M* for one time instead of encrypt for each common users. She will upload the permission assignment to server in cloud and the servers are on duty for data re-encryption. The common user, who wants to access the data *M*, will obtain the *M*

with his only private key instead of different keys for each kind of ciphertext. The system will finish the permission assignment based on the common user's current access condition. In this section, we will analyze the properties of our work by comparing with other works, the detail will be shown in **Table 2**.

Table 2. properties analysis of our work

Properties descriptions	JBE[7]	CPRE[8]	Type-PRE[13]	ACC-PRE[15]	PreBAC
Could be used for ciphertext fine-grained management or not?	No	No	Yes	No	Yes
Could be used for multi-factor or not?	No	Yes	No	No	Yes
Is the users' encryption work difficult or not?	Yes	Yes	Yes	No	No
Is users' keys management work difficult or not?	No	Yes	Yes	No	No

The properties of PreBAC are as follows.

1) Could be used for ciphertext fine-grained management or not?

JBE[7] , CPRE[8] and ACC-PRE[15] have not describe the ciphertext detailed, thus they can not Could be used for ciphertext fine-grained management. Type-PRE[13] and our scheme are suitable for fine-grained management.

2) Could be used for multi-factor or not?

The factors like identity, role,time or environment could be included in the parameter *conU*. CPRE[8] and our scheme could satisfy this requirement.

3) Is the users' encryption work difficult or not?

The work for data re-encryption is finished by cloud server instead of data owner, thus the users' encryption work and cost will be reduced.

4) Is users' keys management work difficult or not?

The user will only need to manage his private key instead of different keys for different ciphertext, thus users' keys management work is much easier.

6.2 Security analysis

1) Security proof

(1) Security Model of PreBAC

We will setup the security model Based on [16] and DBDH problem, adversary A can query the oracles such as first round encryption, key generation, re-encryption key generation, re-encryption, and decryption .

Setup: Challenger setups system parameters $param$.

Phase 1: Adversary A could query one of the oracles including$First_Enc$, $KeyGen$, $ReKeyGen$, $ReEnc$, Dec_1 andDec_2.

During the querying of $First\_Enc$, $ReKeyGen$, $ReEnc$, Dec_1, Dec_2, $A's$ private key is generated by $KeyGen$.

Challenge: After A finishing Phase 1, the challenger picks $m_0, m_1 \in M$, the multi-factor con_U* and public key pk^* which is also generated by $KeyGen$ and corresponding private key is not disclosed. While A is querying $ReKeyGen$ with (pk^*, pk', con_U*), the corresponding

private key of pk' could not be disclosed. Challenger picks $b \in \{0,1\}$ randomly and gets $C_b = First\_Enc(m_b, pk^*)$ as the challenge to A.

Phase 2: A is allowed to query the oracles as similar as Phase 1. While we need the constraints as follows..

a. If A queries $ReKeyGen$ with (pk^*,pk',con_U^*), the private key corresponding with pk' is undisclosed.

b. If A queries $ReEnc$ with (C_b,pk^*, pk',con_U^*) , the private key corresponding with pk' is undisclosed.

c. A cannot query Dec_1 with (C_b,pk^*) directly.

d. If A queries $ReKeyGen$ with (pk^*,pk',con_U^*), A cannot query Dec_2 with C_b', where C_b' is valid.

Guess: A gives a guess $b' \in \{0,1\}$, if $b' = b$,it will success.

Theorem: ***For assumption***, we define the advantage of A to success as ε, and $\varepsilon = |Pr[b' = b] - \frac{1}{2}|$. If ε could be negligible, then A fail, It means that PreBAC is CCA security.

If DBDH assumption holds in groups (G_1,G_2), then ε could be negligible and PreBAC is CCA security based on random oracle model.

(2) Proof scheme

Let us define challenging games set as $\mathcal{G}_i (i = 1, \cdots, 6)$, challenger as B, and T_i as the event which will happen when $b' = b$ in $\mathcal{G}_i$.

(a) $\mathcal{G}_0$: The challenger B faithfully responses the oracle queries from A. Meanwhile, B Setups $H_i^{list} (i = 1, \cdots, 4)$ by selecting $\pi_1, \pi_4 \in G_1, \pi_2 \in Z_p^*, \pi_3 \in \{0,1\}^l$ and setting (pk_i,π_1),(m,k,π_2),(k,π_3), (c_1,c_2 ,c_3 ,c_4 ,π_4) in $H_i^{list} (i = 1, \cdots, 4)$. Let $\delta_0 = Pr[b' = b]$, then $|\delta_0 - \frac{1}{2}| = \varepsilon$.

(b) $\mathcal{G}_1$: Challenger B does in the same as $\mathcal{G}_0$, except the following:

B randomly picks up $\tau \in \{1, 2, \cdots, p+1\}$ to query H_1in τ times. When B receives the challenge from A to queryH_1, B will aborts this game. Thus, the probability of B to succeed is at least$\frac{1}{p+1}$. $\delta_1 = Pr[b' = b]$ in $\mathcal{G}_1$, and then $Pr[T_1] = \frac{\delta_1}{p+1}$.

(c) $\mathcal{G}_2$: Challenger B does as similar as $\mathcal{G}_1$, besides conflicting H_i. For hashes are defined under the random oracles, thus $|Pr[T_1] - Pr[T_2]|$ could be negligible.

(d) $\mathcal{G}_3$: Challenger B does as similar as $\mathcal{G}_2$, besides the query of Dec_2. In the oracle of Dec_2 querying, if the input is (C,pk^* , con_U^*) and A has not queried H_1 with ($pk^* \parallel con_U^*$), then B will abort this game, or B will return the ciphertext to A. Because the hash functions are defined under the standard random oracles and the whole cryptography algorithms are certain, $|Pr[T_2] - Pr[T_3]|$ is also negligible.

(e) $\mathcal{G}_4$:Challenger B does as similar as $\mathcal{G}_3$, besides the querying of Dec_1. If A has not queried H_2 with $m_b \parallel k^*$, there is no differences between $\mathcal{G}_4$ and $\mathcal{G}_3$. Therefore, $|Pr[T_3] - Pr[T_4]|$ could be negligible.

(f) $\mathcal{G}_5$: Challenger B does as similar as $\mathcal{G}_4$, besides the querying of $ReKeyGen$ and $ReEnc$. During this query, B matches re-encryption key list with the condition (pk_i, pk_j, con_U^*)

proposed by A. If there returns a result of this search, then B returns $rk_{i\to j}$ to A, or B will go on as follows.

If user $i's$ private key is corrupted, which means $sk_i = x_i$, then B computes $rk_{i\to j} = (pk_j, pk_j^r, H_1(pk_j||con_U) \cdot H_1(pk_i)^{sk_i}, g^{-r})$

If user $i's$ private key is uncorrupted, then B picks $a \in G_1$, set $sk_i = ax_i$, and compute $rk_{i\to j} = (pk_j, pk_j^r, H_1(pk_j||con_U) \cdot H_1(pk_i)^{sk_i}, g^{-r})$.

If $j's$ private key is corrupted, B will abort.

When querying $ReEnc$, B will compute re-encrypted cipher text by $ReEnc$ with (pk_i, pk_j, C_i) proposed by A. If it does not hold, B aborts. Or, B will search the private keys from the lists of private key and re-encryption key, then he returns cipher text to A. If pk_j is not generated by $KeyGen$, B aborts. $|Pr[T_4] - Pr[T_5]|$ could be negligible.

(g) $\mathcal{G}_6$: Challenger B does as similar as $\mathcal{G}_5$, besidess the following situations.

When B gets $A's$ challenging (m_0, m_1, con_U), B decrypts the cipher text, and then picks $b \in \{0,1\}$ to compute $k \in G_2$, $r = H_2(m_b||k)$, $c_1 = g^r$, $c_2 = k \cdot e(pk_i, H_1(pk_i))^r$, $c_3 = m \bigoplus H_3(k)$, $c_4 = H_1(pk_i)$, $c_5 = H_4(c_1||c_2||c_3||c_4)^r$. Thus, $\mathcal{G}_6$ is different from $\mathcal{G}_5$ based on the querying of H_3. The mathematical complexity of querying H_3 is as similar as the DBDH problem, therefore $|Pr[T_5] - Pr[T_6]|$ could be negligible. All the hash functions are defined under random oracles, thus $Pr[T_6] = \frac{1}{2(p+1)}$. $|Pr[T_1] - Pr[T_6]| = |Pr[T_1] - \frac{1}{2(p+1)}|$ could be negligible by analyzing in (a) to (g), the $\Pr[T_1] = \frac{\delta_0}{p+1}$ and $|\frac{2\delta_0 - 1}{2(p+1)}| = |\frac{\delta_0 - \frac{1}{2}}{(p+1)}| = |\frac{\varepsilon}{(p+1)}|$ could be negligible. Thus, ε could be negligible. And our proof scheme is finished.

2) System security analysis

Our scheme has encrypted information M by symmetric method based on traditional algorithms, which can defense the hacker attacking. And the symmetric *k* for *M* has been encrypted by the algorithm in Section 5.4, which is proved to be CCA-security.

7. Conclusion

With the development of cloud computing, the methods for data sharing and processing have been improved. The individual users could obtain information, software, platform or even infrastructure in the form of cloud services. However, cloud also brings a serious situation to users' privacy protection. Therefore, how to prevent the user's information from being lost and stolen becomes a vital issue for cloud. Access control is still useful for cloud as the other traditional technologies. For the special situations in cloud, how to design an access control scheme for both multi-factor and cryptographic management will a serious problem to be solved in cloud. We have proposed a PreBAC scheme; firstly, we have shown the motivation by an example. And then, we have given the aims and assumptions. Thirdly, the system model, system stages and algorithms have been explained. Finally, we have discuss the properties of PreBAC and proved the scheme. Our work is suitable for cloud computing without increasing the encryption and key management cost of individual user.

Acknowledgment

This work has been supported by the National Natural Science Foundation of China (61702266) and Natural Science Foundation of Jiangsu Province (BK20150787).

References

[1] Y. D. Wang, J. H. Yang, C. Xu, et al, "Survey on Access Control Technologies for Cloud Computing," *Ruan Jian Xue Bao/ Journal of Software*, vol.26, no. 5, pp. 1129-1150, May, 2015. Article (CrossRef Link).

[2] S. Jha, S. Sural and J. Vaidya et al, "Security Analysis of Temporal RBAC under an Administrative Model," *Computers & Security*, vol. 46, pp.154 - 172, Oct. 2014. Article (CrossRef Link).

[3] L. Yang, Z. Tang, R. F. Li, et al, "Roles query algorithm in cloud computing environment based on user require," *Journal of Communications*, vol.32, no.7, pp 169-175, July, 2010. Article (CrossRef Link).

[4] J. Luo, H. Wang and X. Gong, et al. "A Novel Role-Based Access Control Model in Cloud Environments," *International Journal of Computational Intelligence Systems*, vol.9, no.1, pp. 1-9, Feb. 2016. Article (CrossRef Link).

[5] J. Li, A. Squicciarini, D. Lin, et al, "SecLoc: Securing Location-Sensitive Storage in the Cloud," in *Proc. of SACMAT'15*, pp.51-61, June. 2015. Article (CrossRef Link).

[6] L. Zhou, V. Varadharajan, M. Hitchens, "Trust Enhanced Cryptographic Role-Based Access Control for Secure Cloud Data Storage," *IEEE Transactions on Information Forensics and Security*, vol.10, no.11, pp. 2381-2395, Nov. 2015. Article (CrossRef Link).

[7] L. Zhou, V. Varadharajan, K. Gopinath, "A Secure Role-Based Cloud Storage System for Encrypted Patient-Centric Health Records," *The Computer Journal*, vol.59, no.11, pp. 1593-1611, July. 2016. Article (CrossRef Link).

[8] P. Xu, T Jiao, Q. Wu, et al, "Conditional Identity-Based Broadcast Proxy Re-Encryption And Its Application to Cloud Email," *IEEE Transactions on Computers*, vol.65, no.1, pp.66-79, Mar. 2015. Article (CrossRef Link).

[9] Y. Zhang, J. Li, X. Chen, et al, "Anonymous Attribute Based Proxy Re-Encryption for Access Control in Cloud Computing," *Security and Communication Networks*, vol. 9, no.14, pp.2397-2411, July. 2016. Article (CrossRef Link).

[10] J. Li, X. Zhao and Y. Zhang et al, "Provably Secure Certificate-based Conditional Proxy Re-encryption," *Journal of Information Science & Engineering*, vol.32, no.4, pp. 813-830, July. 2016. Article (CrossRef Link).

[11] Q. Liu, G. Wang, J. Wu, "Time-Based Proxy Re-Encryption Scheme for Secure Data Sharing in a Cloud Environment," *Information Sciences*, vol. 258, no.3, pp.355-370, Feb. 2014. Article (CrossRef Link).

[12] Y. Yang, H. Lu and J. Weng et al, "Fine-Grained Conditional Proxy Re-Encryption and Application," in *Proc. of ProvSec 2014*, pp. 206-222, Oct.2014. Article (CrossRef Link).

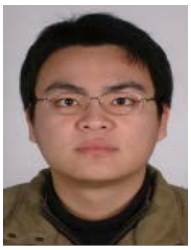
[13] Q. Tang, "Type-Based Proxy Re-encryption and Its Construction," Proc. INDOCRYPT 2008. Springer Berlin Heidelberg. pp. 130-144. 2008. Article (CrossRef Link).

[14] M. Su, G. Z. Shi Z, R. N. Xie, et al, "Multi-element based on proxy re-encryption scheme for mobile cloud computing," *Journal of Communications*, 36(11):73-79, 2015. Article (CrossRef Link).

[15] M. Su, F. H. Li, G. Z. Shi, et al, "A User-Centric Data Secure Creation Scheme in Cloud Computing," *Chinese Journal of Electronics*, vol.25, no.4, pp. 753-760, April, 2016. Article (CrossRef Link).

[16] X. Jia, J. Shao, J. Jing, et al. "CCA-secure type-based proxy re-encryption with invisible proxy," in *Proc. of Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on. IEEE*, 1299-1305, 2010. Article (CrossRef Link).

Mang Su received a B.S. degree in Beijing Electric Science and Technology Institute, China, in 2009, and Ph.D. degree majoring in Cryptography from Xidian University, China, in 2014. Since 2015, she has been a lecturer with the school of computer science and engineering at Nanjing University of Science and Technology, China. Her research interests include cloud security and privacy protection.



Liangchen Wang received a B.S. degree in Xidian University, China, in 2007, and M.S. degree majoring in Cryptography from Xidian University, China, in 2012. Since 2012, he has been a engineer of cryptography at Nanjing Municipal Public Security Bureau, China. His research interests include cryptography and cyber security.