A Privacy Preserving Friend Discovery Strategy Using Proxy Re-encryption in Mobile Social Networks

Entao Luo[1,3], Wenbo Wang[1], Dacheng Meng[1], and Guojun Wang[2](✉)

[1] School of Information Science and Engineering,
Central South University, Changsha 410083, China
csgjwang@gmail.com
[2] School of Computer Science and Educational Software,
Guangzhou University, Guangzhou 510006, China
[3] School of Electronics and Information Engineering,
Hunan University of Science and Engineering, Yongzhou 425199, China

Abstract. In mobile social networks, based on secret sharing and CP-ABE, focusing on the security and privacy issues of friend discovery, we propose a matching scheme under different authorities and realize cross domain data access and sharing. By using proxy re-encryption technology, we hide users' access policy, which can guarantee the security and privacy in the friend making process. Because agents attend encryption and decryption, the privacy can be largely enhanced and the bottleneck of single authority also will be solved. Security and performance analysis show that the relationship of ciphertext's size and access policy is linear, which can resist collusion attack and meet CPA security, our scheme is superior to the existing schemes.

Keywords: Ciphertext-policy access control · Cross domain data access · Proxy re-encryption · Attribute-based encryption · Privacy preserving

1 Introduction

1.1 Background

With the rapid development of mobile social networks (MSN) and intelligent terminal equipment [1–4], users can share their emotions, photos, activities and hobbies to find new friends in MSN, many social applications can help enlarge the social scope (My Life Here, WeChat, etc.). Users can find friends with same interests or certain characteristics in cloud by comparing the personal attribute profile. But in this process, the cloud service provider (CSP) cannot be fully trusted, which may cause security risks of the stored data. For example, CSP may provide users' information to third parties without permission, which will affect users' data security. Hence, typically, users need to encrypt sensitive data to ensure the security and privacy.

G. Wang et al. (Eds.): SpaCCS 2016, LNCS 10066, pp. 190–203, 2016.
DOI: 10.1007/978-3-319-49148-6_17

Attribute-based encryption scheme is a typical application of privacy protection in mobile social networks, including keyPolicy attribute based encryption (KP-ABE) [19–22] and ciphertext policy attribute based encryption (CP-ABE) [23–25]. In KP-ABE, the decryption key is related to the access policy, ciphertext is related to attributes set. If the attributes set in ciphertext can satisfy the access policy in secret key, the data visitor can decrypt the ciphertext. On the contrary, in CP-ABE, data owner can define special access policy depend on personal attribute profile. Secret keys are associated with attributes set, when and only when the attributes in secret keys can satisfy the access policy in ciphertext, users can obtain the plaintext, so data owners can control their data more directly. Hence, comparing to KP-ABE, CP-ABE is more suitable for friend discovery in mobile social networks.

In the system model and working mechanism, the existing modes always depend on single Trusted Authority ($\mathcal{TA}$) to distribute public keys and utilize the access tree generated from user's attribute to achieve access control to other users. But in this kind model, users are working in the same field, that is to say, the generation and distribution of all keys is generated by the same trusted authority.

Obviously, this model is not consistent with the actual application scenarios. For instance, in real dating system environment, data is stored in different clouds, when a data visitor expects to access data and exchange it, it is not possible to expect both data owner and visitor are in the same domain, inter cloud access needs to be taken into account. At the same time, in this model, the user's access control structure exists the risk of violent speculation by malicious attackers, once cracked successfully, it will directly threaten the data privacy. Therefore, single working domain scheme failed. Based on the above problems, this paper considers that users can share data in multi domains by introducing of proxy re encryption technology to ensure the data security.

1.2 Related Work

According to the research on security and privacy protection of friend discovery in the mobile social network, many researchers put forward their research results, literature [9–14] proposed a solution that does not rely on trusted authority, by calculating private set intersection (PSI) to ensure the user's privacy. The main method is: the two matching sides hold their own private attributes sets, by calculating the intersection or the intersection of the cardinal of the two sets to prevent the privacy leakage. Zhang et al. [15] improved the above methods, and proposed to distribute the different weight to the user's interest and calculate the similarity. In the follow-up work, Niu et al. [16] set the user's attributes with priority and improved it. Zhu et al. [17] proposed efficient confusion matrix transform algorithm to achieve a safe and efficient matching.

However, in the above schemes, users can only compare the number and weight of each attribute in the public collection, but do not consider the diversity of user attributes and access control. Therefore, the application range is limited. In the literature [5–8], the security and privacy in the process of making friends can be protected by the introducing the trusted authority and attribute encryption scheme, but in this model the problem of cross domain sharing of user data in the cloud cannot be solve. At the same time, it is a performance bottleneck to rely on single trusted authority. The literature [18] proposed the multi-authority attribute-based encryption scheme with access policy, the protocol using attributes to encrypt the message, and decrypts the message via trusted authority, and provide fine-grained access control to attributes matching and information sharing. But, in this scheme, there is violence speculation risk of access policy tree, once the access policy was successfully guessed, then the attacker can directly decrypt the stored data in a cloud, resulting in a security risk.

Therefore, in order to solve the problem of the performance bottlenecks and violent speculation of access policy, this paper intends to introduce the idea of multi-domain key sharing and proxy re-encryption technologies on the diversification of user management, to ensure the security and privacy of friend discovery in the mobile social networks. The symmetric encryption algorithm is adopted to encrypt the privacy sensitive data of the initiator, then utilize the CP-ABE algorithm to encrypt the symmetric key used in the symmetric encryption algorithm, finally get ciphertext of the key. When responder's attributes satisfy the access policy on initiator, the responder can decrypt the ciphertext of the key to get the decryption key, then decrypt the ciphertext downloaded from friend discovery center to obtain the plaintext. Further social activities can be carried out. The contributions of this paper are as follows:

(1) Based on secret sharing, an access control policy is proposed, ciphertext is associated with access policy, the ciphertext access control structure ensures that users can obtain the correct decryption key in accordance with the requirements of access control structure.
(2) We propose a proxy re-encryption based friend discovery scheme, using proxy re-encryption technology, the access control structure of data owner can be efficiently hided, and the user who satisfies the access control structure can correctly decrypt the encrypted data from the proxy user, which ensures that the friends of proxy user can be efficiently shared and guarantees the privacy of data owner.
(3) A multiple domain encryption scheme based on attributes is proposed, which can realize the data sharing among different domains, expand the scope of making friends, and improve the efficiency of the users.

2 Preliminaries

2.1 Mathematical Basis

Bilinear Mapping: Let $\mathbb{G}$ and $\mathbb{G}'$ be two multiplicative cyclic groups with big prime order p. Let g be a generator of $\mathbb{G}$. Let be a bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}'$ with the following properties:

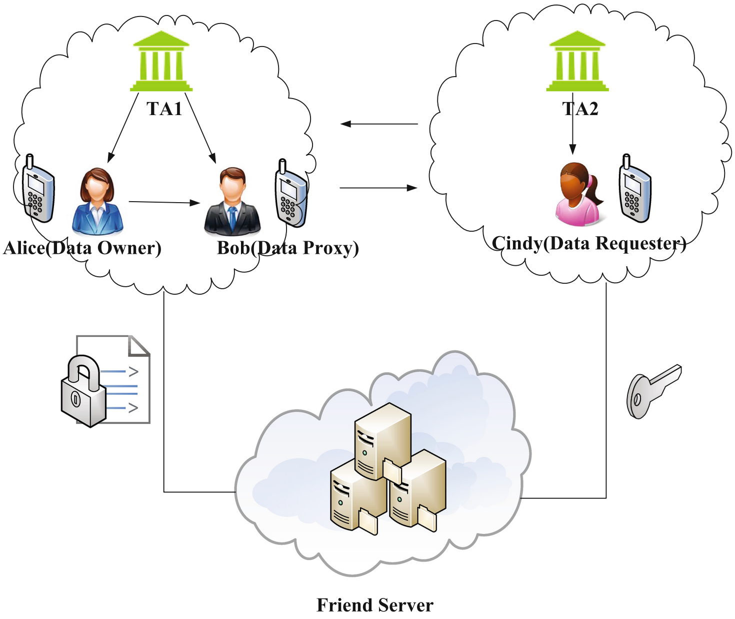

Fig. 1. Friend discovery scheme using proxy re-encryption

(1) Bilinearity: $e(P^a, Q^b) = e(P, Q)^{ab}, \forall P \in \mathbb{G}_0, Q \in \mathbb{G}_1$ and $a, b \in Z_q$.

(2) Non-degeneracy: The mapping will not map all pairs in $\mathbb{G}_0 \times \mathbb{G}_1$ to the identity in $\mathbb{G}_T$, because $\mathbb{G}_0, \mathbb{G}_1$ are groups of prime order, this means that if P and Q are generators of $\mathbb{G}_0$ and $\mathbb{G}_1$, respectively, then $e(P, Q)$ is the generator of $Z \in \mathbb{G}_{\mathbb{T}}$.

(3) Computability: There exists an efficient algorithm to calculate $e(P, Q), \forall P \in \mathbb{G}_0, Q \in \mathbb{G}_1$.

2.2 System Model

The model in this paper mainly consists of the following components: Trusted Authority, Friend Server, Data Owner, Data Proxy, and Data Requester. The model assumes that Trusted Authority, is completely trustworthy and that Friend Server is honest and curious. That is, Friend Server will honestly comply with various system protocols but it will also do what it can to secretly access user files stored in it. Hence, the user should encrypt private files before uploading them to Friend Server. The general structure of the scheme is shown in Fig. 1.

(1) Trusted Authority ($\mathcal{TA}$): Responsible for initializing the system, generating the attribute keys of the region, distributing the keys, and for fine-granularity access control strategies.

(2) Friend Server ($\mathcal{FS}$): Responsible for storing the user's private cipher text, including personal photos, interests, contacts, identifies and private videos.

(3) Data Owner ($\mathcal{DO}$): Responsible for creating, modifying, deleting, encrypting files, and specifying access strategies. The encrypted files cannot be decrypted correctly unless the $\mathcal{DR}$'s property satisfies the $\mathcal{DO}$'s access control strategy before performing further communications. This paper supposes that Alice is $\mathcal{DO}$.

(4) Data Proxy ($\mathcal{DP}$): It is authorized by $\mathcal{DO}$ to re-encrypt $\mathcal{DO}$'s access control structure for the purpose of hiding $\mathcal{DO}$'s actual access control structure. Meanwhile, it can recommend its friends to $\mathcal{DR}$ to improve efficiency of the friend making mechanism. This paper assumes Bob as $\mathcal{DP}$.

(5) Data Requester ($\mathcal{DR}$): Responsible for submitting friend making request to $\mathcal{DP}$. This paper assumes Cindy as $\mathcal{DR}$.

First, $\mathcal{DO}$ uploads the access control strategy of a self-defined property to $\mathcal{TA}$. Each $\mathcal{TA}$ manages the set of properties in their respective domains, generates and distributes private keys for the set of properties owned by users in the domains. To ensure the safety of privacy during the friend making process, $\mathcal{DO}$ needs to encrypt the data to be shared and uploads the encrypted cipher text to $\mathcal{FS}$. During the friend making process, $\mathcal{DO}$ can grant authorization to the proxy, and $\mathcal{DP}$ can recommend friends to $\mathcal{DR}$ that satisfies the proxy's access control structure for the purpose of improving the friend making scope and efficiency of $\mathcal{DR}$, protecting the access control structure of $\mathcal{DO}$ from being intercepted by attackers, and ensuring privacy safety in the friend making process.

The access structure of sensitive data files is specified by $\mathcal{DO}$ or $\mathcal{DP}$. The cipher text of the sensitive data files can be accessed by other $\mathcal{DR}$ that satisfies the access structure. This enables $\mathcal{DO}$ and $\mathcal{DP}$ to flexibly control the access permission of other users.

The proposed scheme security validation relies on the security validation framework based on dual system encryption. The proposed scheme consists of five stages: system initialization phase, user private key generation phase, file encryption phase, cipher text proxy re-encryption phase, and file decryption phase.

3 Details of the Proposed Scheme

3.1 System Initialization Phase

$\mathcal{TA}$ chooses two cyclic groups $\mathbb{G}$ and $\mathbb{G}^{\mathbb{T}}$, whose order is the prime number p. It also randomly chooses elements $g, g_1 \in \mathbb{G}$, $a \in Z_p^*$. Let $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_{\mathbb{T}}$ denote a bilinear mapping. The public parameter $GP = (p, g,\ g_1, g^a, \mathbb{G}, \mathbb{G}_{\mathbb{T}}, e)$, together with the Hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}$ and $H_2 : \mathbb{G}^{\mathbb{T}} \rightarrow Z_p^*$, are generated (Table 1).

Consider that the friend making system has multiple domains D_ϕ. For the TA_{ϕ_i} of any domain D_{ϕ_i}, it can execute the algorithm *setup*( ), randomly choose $\alpha_{\phi_i} \in Z_p^*$, generate the master key of the domain, $MSK_{\phi_i} = g^{\alpha_{\phi_i}}$, and the

Table 1. Summary of notations

Notation	Description
PK_{ϕ_i}, MSK_{ϕ_i}	Public key and master key
SK_S	Private key
KF	Symmetric key
CF	Data ciphertext
CT, CT'	Key ciphertext and re-encrypt key ciphertext
$DataFile$	Data plaintext
$(M,\ \rho)$	Data owner access control structure
$(M',\ \rho')$	Data proxy access control structure
$rk_{S\rightarrow(M',\ \rho')}$	Re-encrypt key
$Setup()$	System initializationfunction
$KeyGen()$	Key generation function
$Enc()$	Encryption function
$ReEnc()$	Re-encryption function
$ReKeyGen()$	Re-encrypt key generation function
$Dec()$	Decryption function

public key $PK_{\phi_i} = e(g,g)^{\alpha_{\phi_i}}$. The public parameter GP and the public key of the domain are public. The master key of the domain, MSK_{ϕ_i}, is stored by TA_{ϕ_i}.

3.2 User Private Key Generation Phase

The user who intends to join the network and participate in social activities, should first initiate the application on the smart terminal, and then chooses to register in a certain TA_{ϕ_i}. The registration process is as follows.

(1) The application of $\mathcal{TA}$ executes the algorithm $keyGen(\ )$, chooses a random number $ts \in Z_p^*$ for the user, and generates the private key $SK_S = (K = g^{a\cdot ts} \cdot g^{\alpha_{\phi_i}}, L = g^{ts}, K_x = H_1(x)^{ts})$.

(2) TA_{ϕ_i} sends (PK_{ϕ_i}, SK_S) and the signature of this user in TA_{ϕ_i} to this user through the safe channels.

3.3 File Encryption Phase

The encryption process of $\mathcal{DO}$ is as follows:

(1) $\mathcal{DO}$ first chooses an unique document number $\mathcal{FID}$ randomly for the document, and randomly generates a symmetric key $\mathcal{KF}$, which is then used to encrypt the plain text $DataFile$ for the purpose of obtaining the cipher text $\mathcal{CF}$.

(2) $\mathcal{DO}$ runs the document encryption algorithm $Enc(\ )$, where (M, ρ) denotes the access control structure of $\mathcal{LSSS}$, M denotes the $l \times n$ matrix, ρ denotes the associated mapping from the rows of M to properties, $\{\rho(i)|1 \leq i \leq l\}$ denotes the property used in the access structure (M, ρ). $\mathcal{DO}$ randomly chooses a secret to be shared, $s \in Z_p^*$, and a vector $v = (s, y_2, ..., y_n), y_2, ..., y_n \in Z_p^*$. $\mathcal{DO}$ also sets $\lambda_i = v \cdot M_i$, where i ranges from i to l, M_i denotes the vector corresponding to the i_{th} row of M. We randomly choose $r_{1,}$ ... $r_l \in Z_p^*$ to compute the cipher text:

$$\begin{aligned} &A_1 = KFile \cdot e(g,g)^{\alpha \cdot s}, A_2 = g^s, A_3 = g_1^s; \\ &B_1 = (g^a)^{\lambda_1} \cdot H_1(\rho(1))^{-r_1}, ..., B_l = (g^a)^{\lambda_l} \cdot H_1(\rho(l))^{-r_t}; \\ &C_1 = g^{r_1}, ..., C_l = g^{r_l}; \end{aligned} \tag{1}$$

The cipher text of the key can be expressed as:

$$CT = ((M, \rho), A_1, A_2, A_3, (B_{1,} C_1), ..., (B_{l,} C_l)) \tag{2}$$

(3) $\mathcal{DO}$ sends (FID, CT, CF) and the signature to $\mathcal{FS}$, which will verify the signature after receiving it. If the signature is valid, (FID, CT, CF) will be stored.

3.4 Cipher Text Proxy Re-encryption Phase

The cipher text proxy re-encryption phase is as follows:

(1) Consider that the user Bob is a validly authorized proxy user that satisfies the access control structure (M, ρ) of $\mathcal{DO}$. Then, after receiving the permission from $\mathcal{DO}$, Bob will execute the algorithm $rekeyGen(\ )$.
Bob inputs the private key $SK = (K, L, K_x)$ and the set of properties S to generate a new access control structure $(M',\ \rho')$, where M' is the $l' \times n'$ matrix, ρ' is the associated mapping from the rows of M to properties. Let $\{\rho'(i)|1 \leq i \leq l'\}$ denote the properties used in the access structure $(M',\ \rho')$.

(2) Bob randomly chooses $s' \in Z_p^*$ and the vector $v' = (s', y_2', ..., y_n'), y_2', ..., y_n' \in Z_p^*$. For i ranging from 1 to l', Bob sets $\lambda_i' = v' \cdot M_i'$, where M_i' is the vector corresponding to the i_{th} row of the matrix M'.

(3) If Bob and Cindy belong to the same $\mathcal{TA}$, D_{ϕ_i}, then Bob randomly chooses $\delta \in G_T$ to compute the cipher text.

$$\begin{aligned} &A_1' = \delta \cdot e(g,g)^{\alpha_{\phi_i} \cdot s'}, A_2' = g^{s'}; \\ &B_1' = (g^a)^{\lambda_1'} \cdot H_1(\rho(1))^{-r_1'}, ..., B_l' = (g^a)^{\lambda_l'} \cdot H_1(\rho'(l'))^{-r_t'}; \\ &C_1' = g^{r_1'}, ..., C_{l'}' = g^{r_l'}; \end{aligned} \tag{3}$$

The cipher text can be expressed as:

$$C'_{(M',\ \rho')} = (A_1', A_2', B'_{1,} C_1', ..., B'_{l,}, C_l') \tag{4}$$

(4) If Bob and Cindy do not belong to the same $\mathcal{TA}$ (e.g., Bob belongs to D_{ϕ_i} and Cindy belongs to D_{ϕ_j}), Bob will apply for the public key $e(g,g)^{\alpha_{\phi_j}}$ of the domain D_{ϕ_j} and compute the cipher text.

$$\begin{aligned} &A_1' = \delta \cdot e(g,g)^{\alpha_{\phi_j} \cdot s'}, A_2' = g^{s'}; \\ &B_1' = (g^a)^{\lambda_1'} \cdot H_1(\rho(1))^{-r_1'}, ..., B_l' = (g^a)^{\lambda_l'} \cdot H_1(\rho'(l'))^{-r_t'}; \\ &C_1' = g^{r_1'}, ..., C_{l'}' = g^{r_l'}; \end{aligned} \tag{5}$$

The cipher text can be expressed as:

$$C'_{(M',\ \rho')} = (A_1', A_2', B'_{1,} C_1', ..., B'_{l,}, C_l') \tag{6}$$

(5) Bob chooses $\theta \in Z_p^*$ and computes:

$$\begin{aligned} &rk_1 = K^{H_2(\delta)} \cdot g_1^{\theta} = (g^{a \cdot ts} \cdot g^a) g_1^{\theta}, rk_2 = g^{\theta}, rk_3 = L^{H_2(\delta)}, \\ &\forall x \in S, rk_4 = C'_{(M',\ \rho')}, R_x = K_x^{H_2(\delta)} \end{aligned} \tag{7}$$

Bob outputs the re-encrypted key:

$$rk_{S \to (M',\ \rho')} = (S, rk_1, rk_2, rk_3\ , rk_4, R_x) \tag{8}$$

and sends $rk_{S \to (M',\ \rho')}$ to $\mathcal{FS}$.

(6) After receiving $rk_{S \to (M',\ \rho')}$, $\mathcal{FS}$ re-encrypts the cipher text of the key using the algorithm *reEnc*( ) and outputs the re-encrypted cipher text of the key, CT'. The calculation process is as follows:

If $I \subset \{1, ..., l\}$ is defined as $I = \{i : \rho(i) \in S\}$, $\{\lambda_i\}$ denotes the valid sharing of the secret s based on the matrix M, and S satisfies (M, ρ), then there exists a set of constants $\{\omega_i \in Z_p^*\}_{i \in I}$ which has $\sum_{i \in I} \omega_i \cdot \lambda_i = s$. Afterwards, we compute:

$$A_4 = \frac{e(A_2, rk_1)/e(A_3, rk_2)}{(\prod_{i \in I} (e(B_i, rk_3) \cdot e(C_i, R_{\rho(i)}))^{w_i})} \tag{9}$$

Output:

$$CT' = ((M',\ \rho'), A_1, A_3, (B_{1,} C_1), ..., (B_{l,} C_l), A_4, rk_4) \tag{10}$$

3.5 Document Decryption Phase

The document decryption phase is as follows:

Cindy issues a request to $\mathcal{FS}$ to access the encrypted document $\mathcal{CF}$ with a document number $\mathcal{FID}$. If the set of properties of Cindy, $\mathcal{S}$, does not satisfy (M, ρ), then output the empty set $\perp$. If $\mathcal{S}$ satisfies $(M, \rho$, Cindy can download the encrypted *DataFile* of $\mathcal{DO}$. Hence, Cindy needs to use the decryption algorithm *Desc*( ) to decrypt the cipher text of the key. The steps are as follows:

(1) If the cipher text of the key is the original cipher text $\mathcal{CT}$:
Then define $I \subset \{1, ..., l\}$ as $I = \{i : \rho(i) \in S\}$. There exists a set of constants $\{\omega_i \in Z_p^*\}_{i \in I}$ which has $\sum_{i \in I} \omega_i \cdot \lambda_i = s$. Cindy computes:

$$\begin{aligned} A_4 &= \frac{e(A_2, rk_1)/e(A_3, rk_2)}{(\prod_{i \in I} (e(B_i, rk_3) \cdot e(C_i, R_{\rho(i)}))^{w_i})} \\ &= \frac{KF \cdot e(g,g)^{\alpha \cdot s} (\prod_{i \in I} (e(g^{\alpha \cdot \lambda_i} \cdot H_1(\rho(i))^{-r_i}, g^{ts}) \cdot e(g^{r_i}, H_1(\rho(i)^{ts}))^{w_i})}{e(g^s, g^{a \cdot ts} \cdot g^{\alpha})} \\ &= \frac{KF \cdot e(g,g)^{\alpha \cdot s} e(g, g^{a \cdot ts})^{\sum_{i \in I} \lambda_i \cdot w_i}}{e(g^s, g^{a \cdot ts} \cdot g^{\alpha})} \\ &= \frac{KF \cdot e(g,g)^{\alpha \cdot s} e(g, g^{a \cdot ts})^{\sum_{i \in I} \lambda_i \cdot w_i}}{e(g,g)^{\alpha \cdot s}} \\ &= KF \end{aligned} \tag{11}$$

(2) Consider the case where the cipher text of the key is the re-encrypted cipher text of the key:

a. If $I' \subset \{1, ..., l'\}$ is defined as $I' = \{i : \rho'(i \in S'\}$ and $\{\lambda'_1\}$ is defined as the valid sharing of the secret s' based on M', then there exists a set of constants, $\{w'_i \in Z_p^*\}_{i \in I^*}$, which has $\sum_{i \in I} w'_i \cdot \lambda'_i = S'$. The user Cindy computes δ as:

$$\delta = A'_1/e(A'_2, K')/(\prod_{i \in I} (e(B'_i, L') \cdot e(C'_i, K'_{\rho(i)}))^{w_i})) \tag{12}$$

Correctness validation 1: If Cindy and Bob belong to the same domain D_{ϕ_i}:

$$\begin{aligned} &A'_1/e(A'_2, K')/(\prod_{i \in I} (e(B'_i, L') \cdot e(C'_i, K'_{\rho(i)}))^{w'_i})) \\ &= \frac{\delta \cdot e(g,g)^{\alpha_{\phi_i} \cdot S'} (\prod_{i \in I} (e(g^{a \cdot \lambda'_i} \cdot H_1(\rho'(i))^{-r'_i}, g^{t_{S'}}) \cdot e(g^{r'_i}, H_1(\rho'(i))^{t_{S'}}))^{w'_i})}{e(g^{S'}, g^{a \cdot t_{S'}} \cdot g^{\alpha_{\phi_i}})} \\ &= \delta \end{aligned} \tag{13}$$

If Cindy and Bob does not belong to the same domain (e.g., Bog belongs to D_{ϕ_i} and C belongs to D_{ϕ_j}):

$$\begin{aligned} &A'_1/e(A'_2, K')/(\prod_{i \in I} (e(B'_i, L') \cdot e(C'_i, K'_{\rho(i)}))^{w'_i})) \\ &= \frac{\delta \cdot e(g,g)^{\alpha_{\phi_j} \cdot S'} (\prod_{i \in I} (e(g^{a \cdot \lambda'_i} \cdot H_1(\rho'(i))^{-r'_i}, g^{t_{S'}}) \cdot e(g^{r'_i}, H_1(\rho'(i))^{t_{S'}}))^{w'_i})}{e(g^{S'}, g^{a \cdot t_{S'}} \cdot g^{\alpha_{\phi_j}})} \\ &= \delta \end{aligned} \tag{14}$$

b. Compute the cipher text of the key
$KF = A_1/(A_4)^{\frac{1}{H_2(\delta)}}$, and $A_4 = \frac{e(A_2, rk_1)/e(A_3, rk_2)}{(\prod_{i \in I} (e(B_i, rk_3) \cdot e(C_i, R_{\rho(i)}))^{w_i})}$.
Correctness validation 2:

$$
\begin{aligned}
A_4 =& \frac{e(A_2, rk_1)/e(A_3, rk_2)}{(\prod_{i \in I} (e(B_i, rk_3) \cdot e(C_i, R_{\rho(i)}))^{w_i})} \\
=& \frac{e(g^S, (g^{a \cdot t_S} \cdot g^{\alpha_{\phi_i}})^{H_2(\delta)} \cdot g_1^{\theta})/e(g_1^S, g^{\theta})}{(\prod_{i \in I} (e((g^a)^{\lambda_i} \cdot H_1(\rho(i))^{-r_i}, (g^{t_S})^{H_2(\delta)}) \cdot e(g^{r_i}, H_1(\rho(i))^{t_S \cdot H_2(\delta)}))^{w_i}} \\
=& \frac{e(g^S, (g^{a \cdot t_S} \cdot g^{\alpha_{\phi_i}})^{H_2(\delta)})/e(g_1^S, g^{a \cdot t_s \cdot H_2(\delta)})}{e(g, g^{a \cdot t_S \cdot H_2(\delta)})^{\sum_{i \in I} \lambda_i \cdot w_i}} \\
=& e(g^S, g^{a \cdot t_{\phi_i} \cdot H_2(\delta)})
\end{aligned}
\tag{15}
$$

$$
A_1/(A_4)^{\frac{1}{H_2(\delta)}} = KF \cdot e(g, g)^{\alpha_{\phi_i} \cdot S}/e(g^S, g^{\alpha_{\phi_i}}) = KF \tag{16}
$$

(3) Finally, Cindy can obtain the data document $DataFile$ by decrypting $\mathcal{CF}$ through $\mathcal{KF}$ in order to perform more profound communication. For example, Cindy can acquire $\mathcal{DR}$'s voice bands, videos, contacts and hobbies.

4 Security Analysis

Consider that the decidable $\mathcal{DBDH}$ hypothesis is valid over (G, G_T), then no adversary $\mathcal{A}$ can conquer the proposed scheme using the access matrix (M^*, ρ^*) with a size of $\ell^* \times n^*(\ell^*, n^* \leq q)$.

Definition. Assume that an opponent $\mathcal{A}$ can conquer the proposed scheme in the $\mathcal{CPA}$ game by a margin of $\varepsilon = Adv_A$, then there is at least one polynomial time algorithm which can solve the $\mathcal{DBDH}$ problem by an undeniable margin.

Proof: $\mathcal{A}$ challenger $\mathcal{C}$ is constructed for the decidable $\mathcal{DBDH}$ hypothesis, determining $T = e(g, g)^{a^{q+1} \cdot S}$ or $T \in \mathbb{G}_T$.

$\mathcal{C}$ and $\mathcal{A}$ play the following $\mathcal{CPA}$ game: $\mathcal{C}$ inputs (p, g, G, G_T, e), $\mathcal{DBDH}$ instance $\boldsymbol{y}$ and T, and then determine $T = e(g, g)^{a^{q+1} \cdot S}$ or $T \in \mathbb{G}_T$.

(1) **Initialization phase.** $\mathcal{A}$ delivers the access structure (M^*, ρ^*) to be challenged to $\mathcal{C}$, where M^* is a matrix with a size of $\ell^* \times n^*$, ℓ^* is the number of rows, and n^* is the number of columns $(\ell^*, n^* \leq q)$.

(2) **Establishment phase.** If the property of the access control structure (M^*, ρ^*) belongs to the domain ϕ_i, then $\mathcal{C}$ chooses $\alpha_{\phi_i}, \gamma \in Z_p^*$, sets $g_1 = g^y$, and $e(g, g)^{\alpha_{\phi_i}} = e(g^{\alpha}, g^{\alpha^q}) \cdot e(g, g^{\alpha_{\phi_i}})$. Meanwhile, $\mathcal{C}$ chooses the Hash function H_1, H_2, sends the public parameter $GP = (p, g, G, G_T, e, g_1, g^{\alpha}, H_1, H_2)$ and the public key $PK = e(g, g)^{\alpha_{\phi_i}}$ to $\mathcal{A}$.
$\mathcal{A}$ simulates fulfiment of the random prophecy $H_j (j \in \{1, 2\})$ by establishing the table $H_j^{List} (j \in \{1, 2\})$. And $\mathcal{C}$ answers the queries based on the following rules.
    (a) H_1: $\mathcal{C}$ receives a query H_1 over $x \in U_{\phi_i}$. If the table H_1^{List} has contained the tuple $\{x, z_x, \delta_{2,x}, z_x \in Z_q^*, \delta_{2,x} \in G\}$, $\mathcal{C}$ returns the value $\delta_{2,x}$ in the tuple to $\mathcal{A}$. Otherwise, $\mathcal{C}$ constructs $\delta_{2,x}$. Let X denote the set of labels$\rho^*(i) = x, (1 \leq i \leq \ell^*)$.

$\mathcal{C}$ chooses $z_x \in Z_q^*$, and sets: $\delta_{2,x} = g^{z_x} \cdot \prod_{i \in X} g^{\alpha \cdot M_{i,1}^*/b_i + \alpha^2 \cdot M_{i,2}^*/b_i + \ldots + \alpha^{n^*} \cdot M_{i,n^*}^*/b_i}$.
If X is empty, then $\mathcal{C}$ sets $\delta_{2,x} = g^{z_x}$. $\mathcal{C}$ returns $\delta_{2,x}$ to $\mathcal{A}$ and adds the tuple $(x, z_x, \delta_{2,x})$ to the table H_1^{List}.

(b) H_2: $\mathcal{C}$ receives the query H_2 over $\delta \in G_T$. If H_2^{List} has included the tuple (δ, ξ), $\mathcal{C}$ sends the already included value $\xi \in Z_p^*$ to $\mathcal{A}$. Otherwise, $\mathcal{C}$ sets $H_2(\delta) = \xi$, returns ξ to $\mathcal{A}$, and adds the tuple (δ, ξ) to H_2^{List}.

(3) **Query phase 1.** $\mathcal{A}$ puts a series of queries to $\mathcal{C}$ and $\mathcal{C}$ answers based on the following rules.

(a) The private key extracts the query $O_{SK}(S)$: if $S \vdash (M^*, \rho^*)$, then $\mathcal{C}$ randomly chooses an output from 0,1 and then stops this game. Otherwise, $\mathcal{C}$ chooses a random value $r_S \in Z_p^*$, and finds $w = (w_1, w_2, ..., w_n) \in Z_p^*$, where $w_1 = -1$ and $w\dot{M}_i^* = 0$ when $\forall i, \rho^*(i) \in S$.
If S is in the domain D_{ϕ_i}, then $\mathcal{C}$ sets $L = g^{r_S} \cdot \prod_{i=1,\ldots,n} g^{a^{q+1-i} \cdot w_i} = g^{t_S}$. In this domain, t_S is easily defined as $t_S = r_S + w_1 \cdot a^q + ... + w_n \cdot a^{q-n+1}$. Next, based on this definition, $\mathcal{C}$ constructs $K = g^{\alpha_{\phi_j}} \cdot g^{a \cdot r_S} \cdot \prod_{i=2,\ldots,n} g^{a^{q+2-i} \cdot w_i}$. Validation shows that $K = g^{\alpha_{\phi_j}} \cdot g^{a^{q+1}} \cdot g^{-a^{q+1}} \cdot g^{a \cdot r_S} \cdot \prod_{i=2,\ldots,n} g^{a^{q+2-i} \cdot w_i} = g^{\alpha_{\phi_j}} \cdot L^a = g^{\alpha_{\phi_j}} \cdot g^{a \cdot t_S}$.
If $x \in S$ and $\rho^*(i) \neq x$ for all $i \in \{1, ..., \ell^*\}$, then let $K_x = L^{z_w} = \delta_{2,x}^{t_S} = H_1(x)^{t_S}$.
Otherwise,

$$K_x = L^{z_w} \cdot \prod_{i \in X} \prod_{j=1,\ldots,n} (g^{(a^j/b^j) \cdot r_S} \cdot \prod_{k=1,\ldots,n^*, k \neq j} (g^{a^{q+1+j-k}/b_j})^{w_k})^{M_{i,j}^*} \tag{17}$$

The equation above can prove validity of K_x by using the following equation.

$$\begin{aligned} K_x =& L^{z_x} \cdot \prod_{i \in X} \prod_{j=1,\ldots,n} (g^{(a^j/b_i) \cdot r_S} \cdot \prod_{k=1,\ldots,n^*, k \neq j} (g^{a^{q+1+j-k}/b_i})^{w_k})^{M_{i,j}^*} \\ & \cdot \prod_{i \in X} \prod_{j=1,\ldots,n} (g^{a^{q+1}/b_i})^{w_j \cdot M_{i,j}^*} \\ =& (g^{z_x} \cdot \prod_{i \in X} g^{a \cdot M_{i,1}^*/b_i + a^2 \cdot M_{i,2}^*/b_i + \ldots + a^{n^*} \cdot M_{i,n^*}^*/b_i})^{(r_S + w_1 \cdot a^q + \ldots + w_{n^*} \cdot a^{q-n^*+1})} \\ =& \delta_{2.x}^{(r_S + w_1 \cdot a^q + \ldots + w_{n^*} \cdot a^{q-n^*+1})} \\ =& \delta_{2.x}^{t_S} = H_1(x)^{t_S} \end{aligned} \tag{18}$$

where X is the set of i which has $\rho^*(i) = x$. If S does not satisfy (M^*, ρ^*), then we have $w \cdot M_i^* = 0$.
Hence,

$$\prod_{i \in X} \prod_{j=1,\ldots,n} (g^{a^{q+1}/b_i})^{w_k \cdot M_{i,j}^*} = g^{a^{q+1} \cdot (\sum_{i \in X} \sum_{j=1,\ldots,n^*} w_j \cdot M_{i,j}^*/b_j)} = g^0 = 1 \tag{19}$$

Finally, $\mathcal{C}$ adds the tuple (S, SK_S) to SK^{List}, and returns SK_S to $\mathcal{A}$.

(b) Re-encrypt the key to extract the query $O_{rk}(S,(M',\rho'))$: Use a property set S and an access structure (M',ρ') to query O_{rk}. According to the safety game, if $\mathcal{S}$ does not satisfy (M^*,ρ^*), $\mathcal{C}$ executes $O_{SK}(S)$ first to obtain the corresponding key (K,L,K_x), and then chooses $\theta,\sigma \in_R Z_p^*, \bar{K} \in_R G$. Compute the re-encryption key as $rk_1 = \bar{K} \cdot g_1^{\theta}, rk_2 = g^{\theta}, rk_4 = g^{\sigma}, R_X = \delta_{2,x}^{\sigma}$.

(4) **Challenge stage.** $\mathcal{A}$ outputs m_0, m_1 to $\mathcal{C}$. $\mathcal{C}$ chooses $b \in \{0,1\}$ and answers based on the following rules. For each row i in M^*, set $x^* = \rho^*(i)$ and query H_1 over x^* in order to obtain the tuple $(x^*, z_x, \delta_{2,x^*})$.
Choose $y'_2, y'_3, ..., y'_{n^*}$ and use the vector to share the secret $v = (s, s\cdot a + y'_2, s\cdot a^2 + y'_3, ..., s\cdot a^{n-1} + y'_{n^*}) \in Z_p^{n^*}$. Choose $r'_1, ..., r'_{l^*} \in Z_p^*$, and for all $i \in \{1,2,...,\ell^*\}$, R_i denotes the sets that have $i \neq k$ and $\rho^*(i) = \rho^*(k)$. We define that:

$$B_i^* = \delta_{2,x}^{-r_i} \cdot (\prod_{j=2,\ldots,n} g^{a\cdot M^*_{i,j}\cdot y_j}) \cdot g^{b_i\cdot s\cdot(0z_{x^*})} \cdot (\prod_{k\in R_i}\prod_{j=1,\ldots,n^*} (g^{a^j\cdot s\cdot(b_i/b_k)})^{M^*_{k,j}})^{-1} \tag{20}$$

$$C_i^* = g^{r_i^* + s\cdot b_i} \tag{21}$$

(a) $\mathcal{C}$ chooses $A_1^* \in \{0,1\}^{2k}$, defines $T \cdot e(g^s, g^{\alpha_{\phi_i}}) = A_1^*/m_b$ in an implicit manner and sets $A_2^* = g^s, A_3^* = g_1^s$.

(b) Output the challenging cipher text:

$$CT^* = ((M^*,\rho^*), A^*_{?}, A_2^*, A_3^*, (B_1^*, C_1^*), ..., (B_{\ell^*}^*, C_{\ell^*}^*)) \tag{22}$$

to $\mathcal{A}$. If $T = e(g,g)^{a^{q+1}\cdot s}$, then CT^* is a valid cipher text.

(5) **Query stage 2.** Query as in the first stage but the constraint in Definition 1 needs to be satisfied.

(6) **Prediction stage.** $\mathcal{A}$ outputs a predicted bit $b' \in \{0,1\}$. Then, $\mathcal{C}$ makes its prediction based on the prediction of $\mathcal{A}$. If $\mathcal{A}$ predicts correctly that $b' = b$, then $\mathcal{C}$ outputs the prediction $1(T = e(g,g)^{a^{q+1}\cdot s})$ in the challenge process of the game. Otherwise, $\mathcal{C}$ outputs $0(T \in G_T)$. The success probability of $\mathcal{C}$ can be computed follows.
If the output is 1, i.e. $T = e(g,g)^{a^{q+1}\cdot s}$, then what $\mathcal{A}$ obtains is a valid cipher text about m_b. According to the definition, $\mathcal{A}$ can correctly predict the result. Hence, $\Pr[b' \neq b | (y, T = e(g,g)^{a^{q+1}\cdot s}) = 0] = \frac{1}{2} + Adv_A$.
If the output is 0, i.e. $T \in G_T$, then $\mathcal{A}$ obtains no message on m_b. Hence, the prediction is right at a probability of $\Pr[b' \neq b | (y, T = R) = 0] = \frac{1}{2}$. In this case, $\mathcal{C}$ has an non-negligible advantage of $\frac{\varepsilon}{2}$ in the delidable$\mathcal{DBDH}$ game.

5 Conclusion

In mobile social networks, maximizing the contact and communication between each other, while protecting the privacy of users is a research hotspot in privacy preserving field. Based on cryptography, we propose cross domain re-encryption protocol for privacy preserving. The scheme improves the efficiency of making friends in mobile social networks and enables users find friends satisfying the access policy with fine-grained access control. By using proxy re-encryption, the real access control structure is hidden. The security and privacy of friend discovery in mobile social networks is realized. Meanwhile, we introduce multi-authority, secret keys are generated from several authorities, which solves the bottleneck of single point and key management. From the security analysis, it is proved that the proposed scheme can meet CPA security.

Acknowledgments. This work is supported in part by the National Natural Science Foundation of China under Grant Numbers 61632009, 61472451, 61272151. Hunan Provincial Natural Science Foundation of China under Grant Numbers 2016JJ305. High Level Talents Program of Higher Education in Guangdong Province under Funding Support Number 2016ZJ01. The Fundamental Research Funds for the Central Universities of Central South University 2016zzts060. The Hunan Provincial Education Department of China under grant number 2015C0589.

References

1. Guo, L., Zhang, C., Sun, J., et al.: A privacy-preserving attribute-based authentication system for mobile health networks. IEEE Trans. Mob. Comput. **13**(9), 1927–1941 (2014)
2. Colman-Meixner, C., Develder, C., Tornatore, M., et al.: A survey on resiliency techniques in cloud computing infrastructures and applications. IEEE Commun. Surv. Tutorials **18**(3), 2244–2281 (2016)
3. Luo, E., Liu, Q., Wang, G.: Hierachical multi-authority and attribute-based encryption friend discovery scheme in mobile social networks. IEEE Commun. Lett. **20**(9), 1772–1775 (2016)
4. Xu, Q., Su, Z., Guo, S.: A game theoretical incentive scheme for relay selection services in mobile social networks. IEEE Trans. Veh. Technol. **65**(8), 6692–6702 (2015)
5. Salih, R.M., Lilien, L.T.: Protecting users' privacy in healthcare cloud computing with APB-TTP. In: IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops), 2015, pp. 236–238 (2015)
6. Manweiler, J., Scudellari, R., Cox, L.P.: SMILE: encounter-based trust for mobile social services. In: Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 246–255 (2009)
7. Ge, A., Zhang, J., Zhang, R., et al.: Security analysis of a privacy-preserving decentralized key-policy attribute-based encryption scheme. IEEE Trans. Parallel Distrib. Syst. **24**(11), 2319–2321 (2013)
8. Dong, W., Dave, V., Qiu, L., et al.: Secure friend discovery in mobile social networks. In: Proceedings of IEEE INFOCOM 2011, pp. 1647–1655 (2011)
9. Freedman, M.J., Nissim, K., Pinkas, B.: Efficient private matching and set intersection. In: International Conference on the Theory and Applications of Cryptographic Techniques, pp. 1–19 (2004)
10. Kissner, L., Song, D.: Privacy-preserving set operations. In: Annual International Cryptology Conference, pp. 241–257 (2005)
11. Sang, Y., Shen, H.: Efficient and secure protocols for privacy-preserving set operations. ACM Trans. Inf. Syst. Secur. **13**(1), 315–326 (2009)

12. Cristofaro, E., Kim, J., Tsudik, G.: Linear-complexity private set intersection protocols secure in malicious model. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 213–231. Springer, Heidelberg (2010). doi:10.1007/978-3-642-17373-8_13
13. Li, M., Cao, N., Yu, S., et al.: Findu: privacy-preserving personal profile matching in mobile social networks. In: Proceedings of IEEE INFOCOM 2011, pp. 2435–2443 (2011)
14. Guo, L., Liu, X., Fang, Y., et al.: User-centric private matching for ehealth networks-a social perspective. In: IEEE Global Communications Conference (GLOBECOM), pp. 732–737 (2012)
15. Zhang, R., Zhang, Y., Sun, J., et al.: Fine-grained private matching for proximity-based mobile social networking. In: Proceedings of IEEE INFOCOM 2012, pp. 1969–1977 (2012)
16. Niu, B., Zhu, X., Liu, J., et al.: Weight-aware private matching scheme for proximity-based mobile social networks. In: IEEE Global Communications Conference (GLOBECOM), pp. 3170–3175 (2013)
17. Zhu, X., Chen, Z., Chi, H., et al.: Two-party and multi-party private matching for proximity-based mobile social networks. In: IEEE 2014 International Conference on Communications (ICC), pp. 926–931 (2014)
18. Zhou, Z., Huang, D., Wang, Z.: Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption. IEEE Trans. Comput. **64**(1), 126–138 (2015)
19. Wang, J., Lang, B.: An efficient KP-ABE scheme for content protection in information-centric networking. In: 2016 IEEE Symposium on Computers and Communication (ISCC), pp. 830–837 (2016)
20. Touati, L., Challal, Y.: Collaborative KP-ABE for cloud-based internet of things applications. In: 2016 IEEE International Conference on Communications ICC, pp. 1–7 (2016)
21. Liu, P., Wang, J., Ma, H., et al.: Efficient verifiable public key encryption with keyword search based on KP-ABE. In: 2014 Ninth International Conference on Broadband and Wireless Computing (BWCCA), pp. 584–589 (2014)
22. Okamoto, T., Takashima, K.: Fully secure functional encryption with general relations from the decisional linear assumption. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 191–208. Springer, Heidelberg (2010). doi:10.1007/978-3-642-14623-7_11
23. Waters, B.: Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. In: International Workshop on Public Key Cryptography, pp. 53–70 (2011)
24. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: IEEE Symposium on Security and Privacy (SP 2007), pp. 321–334 (2007)
25. Ramesh, D., Priya, R.: Multi-authority scheme based CP-ABE with attribute revocation for cloud data storage. In: IEEE 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), pp. 1–4 (2016)