Common Friends Discovery with Privacy and Authenticity

Shin-Yan Chiou
Information and Communication Research Labs
Industrial Technology Research Institute
Chutung, Hsinchu, Taiwan 310, R.O.C.
ansel@itri.org.tw

Shih-Ying Chang∗, Hung-Min Sun∗∗
Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan, R.O.C.
godspeed@is.cs.nthu.edu.tw∗
hmsun@cs.nthu.edu.tw∗∗

Abstract

In this paper, we propose a common friend discovery algorithm considering the privacy of users and the authenticity of friend relationships. The privacy means users' other friends' information does not be leaked except their common friends. The authenticity signifies anyone can not successfully claim he is a friend of someone unless he really is. It has many applications such as playing games by friends, finding talking-topics by strangers, finding introducer of job interview, finding matchmaker of someone you desire to know, etc. We consider its security and matching probability. We also implement the algorithm in two mobile phones to prove that it is workable.

1. Introduction

Recently, social networking services have been received a great deal of attentions. They provide online communities of users for information sharing. They also change the way people communicate and interact. Facebook [3], for instance, is a successful application of social networking services. However, almost all of the applications are centralized and a trusted server is necessary.

This centralized deployment results in some limitations. The users have to connect to the server to use the being controlled data. This brings inconvenience because accessing Internet is not always allowable for all users. All private information of a user is stored in the server, so there is the risk of private information leakage. In addition, each user is only authenticated to the sever, so a user has no capability to verify the information provided by another user. As a result, the issues of centralized deployment are inaccuracy, private information leakage, and lack of information authenticity between users.

In this paper, we propose a novel social network-based application which is non-centralized and provides privacy preservation and information authenticity. The application is to find common friends of two users in their personal devices such as cell phones or PDA. The users can use free personal area network (PAN) such as Bluetooth [2] or Infrared Data Association (IrDA) [11] to communicate with each other without Internet access requirement. The application keeps personal information private and the only information users will share is their common friend after executing the application. Furthermore, it authenticates the exchanged information and avoids forging problems.

The rest of this paper is organized as follows. In Section 2, we introduce preliminary. The detailed construction of the propose protocol is shown in Section 3. The analysis of the proposed protocol is given in Section 4. We analyze the security and matching probability. Finally, we show our implementation and conclude this paper in Section 5 and Section 6, respectively.

2. Preliminary

2.1 Private Matching

Recently, research on private matching receives a lot of attention. Several schemes and applications have been proposed [7, 4]. For example, Freedman and Nicolosi [4] apply private matching to Email service (Reliable Email).

The definitions are as follows. Assuming there are two databases $A$ and $B$, one query $Q \subseteq A$ and one matching protocol computes $P = Q \cap B$. The scheme is secure and preserving privacy if it satisfies the following requirements [7].

1. Privacy: Each party can know only $P$ and it’s input to the matching protocol. Except for this information, each party learns nothing.

2. Non-spoofable: Items in database $A$ and $B$ are really authorized by item owners. This means that the
user can make query \( Q \) only if the owners of these queried items authorizes and gives the user these items. In other words, the user can not generate the queried items without authorization of the item owners. Besides that, the user should have some proofs to demonstrate that the item owner indeed authorizes the user.

In Hash Protocol (HP) [7], a person who wants to query the common items in the other’s database computes hash values of items in his own database and so dose the person who is queried. Then they exchange these hash values. By this way, they can find the common items without revealing the information of the un-matched items. On the other hand, Agrawal et al. [1] proposed AgES which uses commutative encryption, instantiating as \( Enc_{k2}(Enc_{k1}(x)) = Enc_{k1}(Enc_{k2}(x)) \), to privately match items.

Also, Freedman et al. [5] proposed a polynomial-based private matching scheme. They use the property of homomorphic encryption provided by Paillier cryptosystem [8] to achieve stronger privacy. A variant of their scheme, set cardinality private matching, let \( A \) know only the set cardinality of \( Q \cap B, |Q \cap B| \), but the actual items in this set. It’s more applicable than previous schemes. After that, Kissner and Song [6] extend FNP scheme to support more functionality. However, these polynomial-based schemes usually have efficiency problem.

Moreover, HP, AgES, and Freedman et al.’s schemes are categorized to asymmetric exchange of information [7], different from symmetric exchange which both parties know the same information in the matching protocol.

Besides those, Li et al. [7] proposed Data Ownership Certificate (DoC) to ensure non-spoofable. DoC provides the authorization of items. If the user does not obtain the item and the corresponding DoC, he can not make the query and convince the other.

2.2 Security Requirements

The security requirements of our proposed protocol are listed as follows.

1. **Privacy preservation.** The protocol reveals nothing but the information of common friends. More precisely, the communicated peer can only know what he already knew. We use the methodology of private matching to design the protocol.

2. **Unspoofability.** The relationship can not be spoofed. More precisely, the relationship claimed by the communicated peer should not be forged. We use the manner similar to data ownership certificate to prevent from data spoofing.

3. Proposed Scheme

In this paper, we proposed a new data matching protocol for finding common friends. We extend HP and DoC [7] to ensure privacy and unspoofability, respectively.

3.1 Working Procedures

We summarize two phases of the proposed protocol as follows.

1. **Phase A: Credential exchange.** In this phase, friends exchange credentials with each other. The credentials include data used in matching and the proofs which show the ownership of the data.

2. **Phase B: Data matching.** In this phase, the common friends are discovered. Additionally, the data ownership, which provides the evidence of being someone’s friend, is also checked.

3.2 The detailed construction of the proposed protocol

3.2.1 Phase A Credential exchange

As shown in Fig. 1, \( Andy \) and \( Gary \) exchange their credentials in this phase. The detailed steps are listed as follows.

1. \( Andy \rightarrow Gary, id_{Andy}||pk_{Andy}||sn_{Andy} \), where \( id_{Andy} \) is the identity of \( Andy \), \( pk_{Andy} \) is the public key of \( Andy \), \( sn_{Andy} \) is a sufficient long string chosen by \( Andy \), and “||” denotes concatenation. The underlying public key cryptosystem can be RSA cryptosystem [10]. Note that key pair \( pk \) and \( sk \) are generated by each user, not a trust third party like CA.

2. \( Gary: computes \( Sig_{Gary}^{Andy} = Sig_{sk_Gary}(h(id_{Gary}||pk_{Andy})) \) for proving he has assigned the ownership to \( Andy \), where \( h(\cdot) \) is cryptographic hash functions.
such as SHA1 [12] or MD5 [9], $sk_{Gary}$ is the private key of Gary, and $Sig_{sk_{Gary}}(\cdot)$ represents the signature signed using key $sk_{Gary}$.

3. Gary $\rightarrow$ Andy, id$_{Gary}$, $pk_{Gary}$, $sn_{Gary}$, $Sig_{Andy}$

4. Andy : verifies $Sig_{Andy}$. If the result is correct, he computes $Sig_{Andy} = Sig_{sk_{Andy}}(h(id_{Andy}) || pk_{Gary})$.

5. Andy $\rightarrow$ Gary, $Sig_{Gary}$

Finally, each peer stores the other’s (id, $pk$, Sig, sn).

3.2.2 Phase B Data matching

In this phase, as shown in Fig. 2, Andy and Bob privately match their common friends, that are denoted as a set of $dm_i$. The steps are shown as follows.

1. Andy $\rightarrow$ Bob, $dm_{Andy}^1||dm_{Andy}^2||...||dm_{Andy}^n||r_A$, where $dm_{Andy}^i = h(id_i || sn_i)$, $n_{Andy}$ is the number of friends haven by Andy, and $r_A$ is a random number. Note that id$_i$ and sn$_i$ are retrieved from Andy’s database.

2. Bob: compares $dm_{Andy}^i$ with $dm_{Bob}^i$ to find the matching items, where $i = 1, 2, ..., n_{Andy}$, $j = 1, 2, ..., n_{Bob}$, and $dm_{Bob}^j = h(id_j || sn_j)$. Note that id$_j$ and sn$_j$ are retrieved from Bob’s database. If $dm_{Andy}^i$ is identical to $dm_{Bob}^j$, this means that Andy and Bob have a common friend. Assuming the number of matching items is nc.

3. Bob $\rightarrow$ Andy, $dm_{Bob}^1||Sig_{Bob}^1||...||dm_{Bob}^n||Sig_{Bob}^n||pk_{Bob}||Sig_{sk_{Bob}}(r_A)||r_B$. In this step, Bob sends nc pairs of $dm$ and $Sig$ which associate to the common friends to prove the data ownerships. Additionally, $pk_{Bob}$ and $Sig_{sk_{Bob}}(r_A)$ are used to show the ownership of $sk_{Bob}$.

4. Andy: verifies the nc signatures for common friends and $Sig_{sk_{Bob}}(r_A)$.

5. Andy $\rightarrow$ Bob, $Sig_{Andy}^1||...||Sig_{Andy}^n||pk_{Andy}||Sig_{sk_{Andy}}(r_B)$.

Finally, Andy and Bob discover the names of nc signatures and $Sig_{sk_{Andy}}(r_B)$. In this step, similar as step 3, Andy prove his ownership to Bob.

6. Bob: verifies the correctness of $Sig_{Andy}$.

Finally, Andy and Bob discover the names of their common friends. Notice that, except their common friends, there is not any information disclosed. Observe that a trust centralized server is unnecessary in our protocol.

4. Analysis of Proposed Scheme

4.1 Security Analysis

We analyze our protocol according to the requirements defined in Section 2.2.

- Privacy preservation. Since $dm_{Andy}^i$ is the hash value of id$_i$ and sn$_i$, Bob does not know the meaning of $dm_{Andy}^i$ unless he has the same pair of id$_i$ and sn$_i$, and neither does Andy. Therefore, the information of their non-common friends keeps private.

- Unspoofability. Since both Andy and Bob have to provide the signature $(Sig_{cf}^i, Sig_{sk_{Andy}}(r_B))$ and $(Sig_{cf}^j, Sig_{sk_{Bob}}(r_A))$ to prove they are friends of cf, where cf is assumed being their common friend, neither Andy nor Bob can spoof the friend relationship without cf’s signature and his own private key.

4.2 Matching Probability

If the matching rate is too low, our applications will lead to impractical. We estimate the probability as follows.

$$P_m = 1 - \binom{N-n}{n}/\binom{N}{n},$$ (1)
where \( N \) is the total number of users in our application and \( n \) is the average number of friends for one user. Fig. 3 shows the matching probabilities for different \( N \). The result shows the probability is acceptable while \( n \) is large enough. We think our protocol may be potential because it is the future trend that the social network of each user becomes larger.

5. Implementation

In this work, we implement a simulation prototype based on our proposed schemes. Fig. 4 shows the mobile phone screens of our implementing prototype. We use C# program language to implement proposed scheme. The types of equipments are HTC Touch Diamond P3700 and P3702. It uses Windows Mobile 6.1 Professional operation system. The technological specifications of the equipment are 256MB ROM, 192MB DDR SDRAM, Qualcomm MSM7201A™ 528 MHz CPU, and Bluetooth 2.0 [2] with EDR.

To implement our proposed scheme, we use two cell phones. The transmission interface is Bluetooth 2.0. According to our proposed scheme, the prototype has two primary capabilities. First, we implement phase A, credential exchange. Any two persons can exchange their credential to each other by using their cell phones. Second, we implement phase B, data matching. Anyone can discover their common friends with other persons by using the credentials exchanged in phase A. If they have common friends, the system shows the names of friends they both know.

Fig. 4 (A) illustrates the initialization screen of our program in mobile phones. Fig. 4 (B) and (C) show the pictures that two cell phones are executing credential exchange. Fig. 4 (D) illustrates two cell phones discovering a common friend, called Gary, after running phase B.

6. Conclusions

In this paper, we present a common friend discovery algorithm considering the privacy of users and the authenticity of friend relationships. The matching probability is discussed to prove the probability is acceptable and our method is practical. Our implementation on two mobile phones is also shown to prove it workable. Therefore, we believe this technology can be directly used in the industry.

References