This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TDSC.2014.2355824

A Trust-based Privacy-Preserving Friend Recommendation Scheme for Online Social Networks

Linke Guo, *Member, IEEE,* Chi Zhang, *Member, IEEE* and Yuguang Fang, *Fellow, IEEE*

Abstract—Online Social Networks (OSNs), which attract thousands of million people to use everyday, greatly extend OSN users' social circles by friend recommendations. OSN users' existing social relationship can be characterized as 1-hop trust relationship, and further establish a multi-hop trust chain during the recommendation process. As the same as what people usually experience in the daily life, the social relationship in cyberspaces are potentially formed by OSN users' shared attributes, e.g., colleagues, family members, or classmates, which indicates the attribute-based recommendation process would lead to more fine-grained social relationships between strangers. Unfortunately, privacy concerns raised in the recommendation process impede the expansion of OSN users' friend circle. Some OSN users refuse to disclose their identities and their friends' information to the public domain. In this paper, we propose a trust-based privacy-preserving friend recommendation scheme for OSNs, where OSN users apply their attributes to find matched friends, and establish social relationships with strangers via a multi-hop trust chain. Based on trace-driven experimental results and security analysis, we have shown the feasibility and privacy preservation of our proposed scheme.

Index Terms—Privacy, Online Social Networks, Trust, Social Relationship

1 INTRODUCTION

Online Social Networks (OSNs) provide people with an easy way to communicate with each other and make new friends in the cyberspace. Similar to what people usually do in real life, OSN users always try to expand their social circles in order to satisfy various social demands, e.g., business, leisure, and academia. In such cases, OSN users may ask for the help from their existing friends to obtain useful feedback and valuable recommendations, and further establish new connections with friends of friends (FoFs). As several works [1], [2] indicates, the social relationship on the OSNs is an asymmetric context-aware trust relationship between two friends, by which we consider the possibility of establishing a multi-hop trust chain two strangers by using existing 1-hop trust of existing friends on the OSNs. However, the recommendation process poses several crucial privacy breaches in the cyberspace, such as OSN users' privacy concerns regarding their identities and social relationships, as well as the recommended information during the information exchange, all of which should be well addressed. Otherwise, it would be very easy for malicious users to perform serious cyber and physical attacks, such as identity theft [3], [4], inferring attack on social relationships [5], and profile leakage [6].

We consider an example that Alice wants to find a cardiologist over some professional OSNs, such as PatientLikeMe[1], for helpful suggestions and preliminary diagnosis. On the one hand, directly asking recommendations to strangers or a non-close friend not only reveals Alice's identity, but also reveals her health condition and medical information. Even worse, traditional recommendation approaches [7], [8] applying identity to recommend strangers will disclose OSN users' social relationships to the public, which impede patients from utilizing it, and also decrease the possibility of establishing the multi-hop trust chain if one of OSN users on the chain returns a negative result. On the other hand, current approaches cannot achieve the fine-grained and context-aware results automatically, due to the fact that OSN users have to determine the recommended friends based on their own judgements on the recommendation query. As in our example, Alice would like to ask for help from her friends who work in a hospital, but not a truck driver. To overcome the above issue, we consider the possibility of using OSN users' social attributes to establish the multi-hop trust chain based on each context-aware 1-hop trust relationship, where most of trust relationships are formed and strengthened by the shared social attributes.

In this paper, we design a light-weighted privacy-preserving friend recommendation scheme for OSNs by utilizing both users' social attributes and their existing trust relationships to establish a multi-hop trust chain between strangers. In our scheme, we jointly consider privacy leakages and preservation approaches regarding the identity, social attributes, and their trust relationships of OSN users during the recommendation process. By trace-driven experimental results, we demonstrate both the security and efficiency of our proposed scheme.

Related Works:

Privacy Issues in OSNs: Several existing works [3], [9], [10] point out the potential security breaches on the OSNs, where they consider adversaries's attack to OSN users'

- *L. Guo is with the Department of Electrical and Computer Engineering, Binghamton University, State University of New York, Binghamton, NY, 13902, USA.*
  E-mail: lguo@binghamton.edu
- *Y. Fang is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, 32611, USA.*
  E-mail: fang@ece.ufl.edu
- *C. Zhang is with School of Information Science and Technology, University of Science and Technology of China, Hefei 230026, China.*
  E-mail: chizhang@ustc.edu.cn

This work was partially supported by the U.S. National Science Foundation under Grant CNS-1343356 and the Natural Science Foundation of China under Grant 61328208. The work of Chi was also partially supported by the Natural Science Foundation of China under Grants 61202140 and 61328208, by the Program for New Century Excellent Talents in University under Grant NCET-13-0548, and by the Innovation Foundation of the Chinese Academy of Sciences under Grand CXJJ-14-S132.

1. http://www.patientslikeme.com/

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TDSC.2014.2355824

identities, attributes, as well as their social relationships. Fong *et al.* [11] propose an access model that formalize and generalize the privacy preservation mechanism for Facebook. Carminati. *et al.* also propose an access control mechanism for the information sharing in web-based social networks (a.k.a. online social networks) in [12], which jointly considers the relationship type, trust metric, and degree of separation in the policy design. The major difference between their scheme and the work in [11] and ours is that they use the decentralized architecture for the access control, which may incur potential security breaches, like fabricating identity, attributes, and trust information. Along this line of research, Squicciarini *et al.* in [13], [14] use game theory to model the privacy management for content sharing, which has the smiler idea as our design in terms of providing privacy for social profile and attributes. In particular, their work in [14] can provide automatic access policy generation for users profile information. Mislove. *et al.* in [15] discuss the possible inference on user profile based on existing relationships, which also could be a very powerful attack on identifying real identities using user attributes.

Trust Management: Comprehensive surveys [16]–[18] on trust and reputation systems for online service provision and mobile ad hoc networks describes the current trends and development in this area. In the most recent survey [19], Sherchan *et al.* summarize the trust management in social networks into three aspects, trust information collection, trust evaluation, and trust dissemination. They also discuss the propagative property of trust, which can be used to create trust chains. In our scheme, we assume the existence of propagative trust among OSN users as the same as in [2], [20], [21]. In addition, from the same source, we are along the line of discussing the context-specific or context-aware trust between OSN users, so that we can leverage it for establishing multi-hop trust chain for users with specific attributes. Lin *et al.* propose a peer-to-peer architecture for heterogeneous social networks in [22], which allow users from different types of social networks to communicate. The proposed architecture also highlights trust management in different types of professional social networks.

Friend Recommendation: In terms of discovering friendships, Daly and Haahr in [23] discuss the establishment of friendship chains using social attributes. Similarly, Chen and Fong in [24] use trust factor in collaborative filtering (CF) algorithm to recommend OSN users on Facebook, where they analyze the similarity based on users' interests and attributes. One of their following work [25] has the same idea, but try to use data mining approach to gather users' information to input to CF algorithm for recommendation. In [8], Dhekane and Vibber discuss the friend finding problem on the Federated social networks. However, the above works fail to consider users' privacy concerns on both identity and their social attributes.

Privacy-preserving profile matching: Li. *et al.* [26] propose a privacy-preserving personal profile matching schemes for mobile social networks, by using polynomial secret sharing. In [27], Dong *et al.* design a secure friend discovery scheme based on verifiable secure dot product protocol by using homomorphic encryption. Due to their distributed approaches, both of the above schemes lack of the ability to prevent active attacks when users change their attributes to satisfy the query requirements. Our previous papers [28]–[30] also discuss the private matching schemes in eHealth/mHealth systems.

Our Contributions: Our major contributions are summarized as follows:

- We utilize OSN users' social attributes and trust relationship to develop the friend recommendation scheme in a progressive way while preserving the privacy of OSN users' identities and attributes.
- We use OSN users' close friends to establish anonymous communication channels.
- Based on the 1-hop trust relationships, we extend existing friendships to multi-hop trust chains without compromising recommenders' identity privacy.
- Our trust level derivation scheme enables strangers to obtain an objective trust level on a particular trust chain.
- Extensive trace-driven experiment are deployed to verify the performance of our scheme in terms of security, efficiency, and feasibility.

The remainder of this paper is organized as follows. Section 2 introduces our intuitions and preliminaries on the proposed scheme. We describe the system and security objectives in Section 3, along with the adversary model of our scheme. The proposed scheme of the trust-based friend recommendation is presented in Section 4, followed by the scheme evaluation in Section 5 and Section 6. Finally, Section 7 concludes the paper.

2 PRELIMINARIES

2.1 Motivation

We first highlight our motivation on the trust-based multi-hop recommendation process. To expand their social circles or find a particular user, they may use their existing trust-based friendships to help recommend friends. Traditional approaches, like ID-based recommendation, recommend a friend by returning a binary answer, "yes" or "no", which lower the possibility of finding friends of friends (FoFs). From the perspective of social networks, most of this type of schemes will fail to extend friendships more than two hops. To increase the possibility of reaching more FoFs, we may have to establish the multi-hop chain for the recommendation. In corresponding with the observation from sociology, the *homophily phenomenon* [31]–[33], OSN users may have social relationship with each other based on their shared attributes. Contrary to the ID-based recommendation, a viable solution is to use each user's social attributes for the recommendation, which will help OSN users search friends in a progressive way. Thus, our scheme is trying to help OSN users recommend FoFs by the increased number of identical attributes hop-by-hop, and establish a multi-hop trust chain between two unknown users after the recommendation.

2.2 Definitions and Assumptions

2.2.1 *Central Authority*

The central authority (CA) is a fully-trusted infrastructure that stores users' social coordinates in its storage. It is also responsible for system setup and generating public/private key pairs to OSN users in the system. In our scheme, we require an always-online CA to provide the recommendation service.

2.2.2 *Trust Level*

The trust level in our system is defined as the reliability trust [16] with propagative property, and it is a numeric value $T \in [0, 1]$ between pair-wise OSN users, where 0 denotes

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TDSC.2014.2355824

lowest trust level and 1 represents the highest level with full trust, respectively. We will use $T_{u_1.u_2}$ to denote OSN user u_1's trust level on u_2. This property denotes that the end-to-end trust relationship can be derived based on each link value [34]. Note that the trust level in this work is also defined as context-aware trust, in the sense that OSN users will forward recommendation to different friends based on different context-aware queries. Here, the use of context-aware trust is as the same as our basic motivation by using attribute-based recommendation, like forwarding request for searching a doctor to someone who is a nurse.

2.2.3 Roles of OSN users

For the ease of description, OSN users are given different roles in our scheme.

- *Querier*(Q): users who initiate the friend recommendation process.
- *Friend*(F): users who are 1-hop away from each other with established friendships.
- *Recommender*(R): users who are strangers to the querier and willing to help the querier discover anonymous trust chain.
- *Destination user*(D): the one that the querier is looking for.

We note that the roles of friend and recommender will be interchangeable in different stages of our scheme. However, from the aspect of the querier, he/she has only one 1-hop friend on one particular trust chain, but may have multiple recommenders depending on the recommendation results, where they are strangers to the querier. Meanwhile, OSN users can bilaterally communicate with each other only if they are friends, while they fail to exchange information if they are strangers.

2.2.4 Social Coordinates

In our system, each user has a unique vector $\mathcal{A} \in \{0,1\}^n$ to represent his/her social attributes, e.g., age, gender, affiliation, etc, where we name it as social coordinate, and n is the length of the vector. The central authority defines a public attribute set consisting of d attributes, $\{\mathcal{A}_1, \mathcal{A}_2, ...\mathcal{A}_d\}$. In each attribute, CA assigns a unique vector to represent the attribute value, e.g., 0010 denotes the user is a student, while 0100 a faculty. Then, recommenders can use the results of the dot-product of two vectors to determine the similarity on attributes. We assume that users' social coordinates used for comparing the similarity would uniquely identify one particular user. In the following sections, we generally use $\mathcal{A}$ instead of using $\mathcal{A}_i$ to denote an OSN user in our scheme.

3 System Model

3.1 Network Model

We first give a brief introduction to the network model of the proposed scheme. As shown in Fig. 1, apart from OSN users, we have a central authority (CA) which is responsible for parameter distribution. The basic assumption of our network model is that there exists secure communication channels between CA and each OSN user. The secure channels can be set up by some authentication and key exchange schemes [35], or by physically using encrypted phone or email. This assumption guarantees the confidentiality of the information distribution from CA.

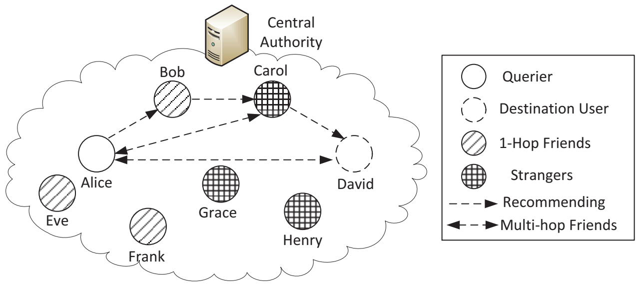


Fig. 1. System Model.

3.2 Design Objectives

Our privacy-preserving friend recommendation scheme should achieve two main objectives:

- **Trust-based Recommendation**
  The multi-hop trust chain can be established by 1-hop trust relationship between pairwise OSN users. Subjective trust levels impact the recommendation performance between two OSN users.
- **Privacy Preservation**
  - *Social coordinate privacy*: Since OSN users are represented as unique sets of social coordinates, directly revealing one's social coordinate vector would leak his/her social privacy and further compromise the identity privacy. We requires that both the recommendation and trust level derivation process cannot reveal OSN users' social coordinates.
  - *Identity and network address privacy*: It requires that the identity and network addresses of both the querier and recommenders will be hidden from each other.
  - *Trust level privacy*: We treat the trust level as private data since it potentially reveals information on friendships and personal social circles. It requires that the trust level between two 1-hop friends cannot be revealed to others during the recommendation process.

3.3 Threat Model

The threat model defines adversaries and their possible attacks to the proposed scheme.

1. **Type I adversary**: They compromise OSN users' identity information and social relationship, and publish to the public domain.
   - The adversary steal OSN users' identity information and further launch attacks to their social relationships and trust levels. To achieve these, they can collect and learn the information regarding the particular trust chains, such as previously used pseudonyms and messages exchanged between friends. Moreover, adversaries can inject bogus data or block users' messages, which tries to prevent the queriers from obtaining the correct aggregated trust level.
2. **Type II adversary**: This type of adversary uses known MAC and IP addresses to track OSN users during the recommendation process.
   - When OSN users recommend friends based on the query, the type II adversary tries to obtain their actual

MAC and IP addresses, and it may to further use this information to locate or track the real identity of particular OSN users.

3. **Type III adversary**: They launch impersonation attacks on honest OSN users and deviate the recommendation process.
   - Adversaries forward recommendation queries to someone that does not satisfy the querier's requirements or even drop the querier's requests. Especially during the trust level derivation process, they can prevent queriers from knowing the correct results.
4. **Type IV adversary**: Adversaries fabricate their own social coordinates and social relationships, which may cause the incorrect recommendation.
   - They will claim they have some required social coordinates as well as particular social relationships with some OSN users who are more similar to the query information. They can also change their social coordinates for malicious purposes, like compromising specific user or obtaining the requirements of some users. In addition, they will compromise honest OSN users' social attributes via their coordinate vectors.

We exclude several attacks according to our design objectives and assumptions. Due to the subjective values on trust levels, we prohibit users from changing it depending on the query and recommendation results. We also exclude the attack launched by a global observer. For a large-scale social network, it is infeasible for a particular user to monitor the whole network except the central authority. Collusion attack is also prohibited in our system because the trust relationships are based on each hop, where users' identities would be revealed if 1-hop friends are involved in the attack.

4 System Design

4.1 Overview

We first give a brief introduction to our proposed scheme. The main design goal of our scheme is to help OSN users securely establish trust relationships with strangers via multi-hop recommendation process. By leveraging existing 1-hop trust relationships, the proposed scheme enables OSN users to extend their social circles while maintaining their identity privacy. For example, imagine Alice(Q), is looking for a cardiologist on a medical OSN as shown in Fig. 1 and Fig.2. However, all of her 1-hop friends (Eve and Frank) do not have the corresponding candidates to recommend. Fortunately, one of her close friends, Bob(F), who worked in a hospital recently, recommends to Alice his best friend Carol(R) for further information. Then, Alice's unknown stranger, Carol, helps her recommend a cardiologist, David(D), who is an acquaintance of Carol. Finally, although Alice and David are strangers before the multi-hop recommendation process, they are connected and form a trust chain via 1-hop friends.

4.2 Privacy-preserving Friendship Establishment

Different from traditional ways to establish friendships, we design a privacy-preserving approach to set up the trust relationships between two OSN users. In what follows, we describe our approach by leveraging users' closest friend sets to enable the communication in a privacy-preserving way.

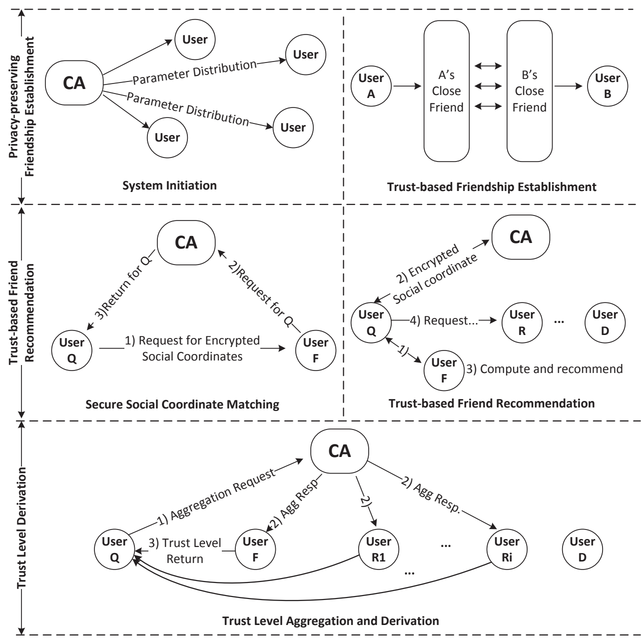


Fig. 2. System Diagram.

4.2.1 System setup

To perfectly hide the identity and network address (IP or MAC address) of an OSN user, we assume each OSN user has a certain number of fully trusted friends, $\mathcal{F}_{ID}^{C}$, which will not reveal any secret information of a user with the particular ID. The assumption satisfies the circumstance in the real life or OSNs with secure channels. For example, people may have several closest friends whom they fully trust $\mathcal{F}_{ID}^{C} \subseteq \mathcal{F}_{ID}$. In order to hide the network address and identity, users can route their packets to the destination via a specific trusted friend, and thereby hide their identities with the help of their close friends.

We first list the notations in Table. 1 and describe system setup of the proposed scheme. Before the scheme runs, CA assigns the ID-based public/private key pairs to each user in the system. The parameter generation procedures are as follows, and we adopt the scheme in [36]:

1. Input the security parameter ξ to the system and output a public parameter tuple (q, G_1, G_2, e, g, H).
2. Randomly select a domain master secret $\varsigma \in Z_q^*$ and calculate the domain public key as $g_{pub} = \varsigma g$.

where e is a bilinear map $e : G_1 \times G_2 \rightarrow G_T$ which has the properties of *bilinearity*, *computability*, and *non-degeneracy* [36], [37]. CA publishes the domain parameters tuple $(q, G_1, G_2, e, g, H, g_{pub})$ and keeps ς confidential, where $H(\cdot)$ is defined before as $H(\cdot) : \{0,1\}^* \rightarrow G_1$, and g is a generator of G_1. Given a specific public $ID \in \{0,1\}^l$, CA distributes the public/private key (pk_{ID}/sk_{ID}) pair as $H(ID)/\varsigma \cdot H(ID)$.

In our scheme, OSN users enable their close friends to use their assigned pseudonyms in the communication instead of real IDs. Similar to the approach proposed in [38], [39], each OSN user assigns a set of collision-resistant pseudonyms to his/her close friends to guarantee the anonymity during recommendation process. We continue to use the previous example to describe the parameter distribution process, and

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TDSC.2014.2355824

TABLE 1
Main Notations

Notation	Description
$\mathcal{F}_u$, $\mathcal{F}_u^C$	Friend set and the closest friend set of a user u
$\mathcal{PS}_{u.i}$	Pseudonym set that the user u assigns to user i
$PS_{u.i}^{\kappa}$	One of user i's pseudonym that the user u assigns, where $1 \leqslant \kappa \leqslant \lvert\mathcal{PS}_{A.i}\rvert$
pk_u/sk_u	User u or pseudonym's public and private key pair
H, $\hat{H}$, H_0	Cryptographic hash function
ς, ς_u	User u's master secret selected by CA, where $\varsigma, \varsigma_u \in Z_p^*$
$\tau_{u_1.u_2}$	Trust level commitment that u_1 evaluates u_2
$\Psi_{u_1.u_2}$	The certificate that user u_1 issues to u_2 for storing u_1's encrypted social coordinates
$\mathcal{C}_{u_1.u_2}$	The credential that u_2 uses to query u_1
$\mathcal{A}$, $\mathcal{Q}$	User's attribute vector and queried vector
$\mathbf{B}_{u1}$, $\mathbf{B}_{u2}$	User u invertible matrices used to generate encrypted social coordinate

for the ease of description, we use Q(querier) and F(1-hop friend) to represent Alice and Bob, respectively. In addition to the scheme in [36], Alice gives to her close friends a set of collision-resistant pseudonyms, $\mathcal{PS}_{Q.i} = \{PS_{Q.i}^{\kappa} | 1 \leqslant \kappa \leqslant |\mathcal{PS}_{Q.i}|, i \in \mathcal{F}_Q^C\}$ and the corresponding private keys as $sk_{\mathcal{PS}_{Q.i}} = \{sk_{\mathcal{PS}_{Q.i}^{\kappa}}\} = \{\varsigma_Q H(PS_{Q.i}^{\kappa}) \in G_1 | 1 \leqslant \kappa \leqslant |\mathcal{PS}_{Q.i}|, i \in \mathcal{F}_Q^C\}$[1]. Note CA distributes the public/private key pairs of the pseudonym sets to each valid OSN user and $\varsigma_Q \in Z_q^*$ is the master secret selected by CA for Alice. Therefore, the close friend $Q.i$ selected by Alice stores the public/private key pairs of a set of pseudonyms for providing anonymity for Alice.

4.2.2 Trust-based Friendship Establishment

Similarly, as a friend of Bob, Alice also obtains a set of pseudonyms to ensure anonymous communication during the recommendation process. However, different from close friends, we require OSN users assign different trust levels $T \in [0, 1]$ to each one of their 1-hop friends and define a map $\mathbb{Q}^+ \to Z_q$ that maps the reliability trust level to an integer on Z_q. We apply Pederson commitment [40] scheme to preserve the trust level between pair-wise OSN users. CA additionally chooses a set parameters (p, g, h) and distributes them to OSN users, where p is a large prime and usually is 1024 bits, $g \in G_1$, $h = g^a \mod p$ and a is a private parameter selected by CA. Once Bob accepts Alice as his friends, he issues her a commitment $\tau_{F.Q} = g^T h^s$ based on the trust level that Bob evaluates on Alice, where $s \in Z_q$ is a random number selected by Bob. Moreover, Bob stores the commitment for responding queries or recommendation requests from Alice, in the sense that Alice or her friends may use pseudonyms to communicate with Bob, but they need to show the commitment so that Bob can ensure the trust relationship established with his 1-hop friend, Alice. Besides, the hiding property of the Pederson commitment scheme guarantee that as a trustee, Alice is not able to uncover the trust level given by Bob.

1. If X is a set, $|X|$ means its cardinality; if X is a number, $|X|$ denotes the length of bits representing the number.

4.2.3 Privacy-preserving Anonymous Communication

After giving the corresponding pseudonym sets to close friends, OSN users within 1-hop can initiate the anonymous communication. For privacy concerns, we design the following scheme to hide users' identities during the recommendation process. Suppose Bob issues Alice several parameters for Alice to contact Bob in the future: $E_{pk_Q}(\mathcal{PS}_{F.Q}), exp, \sigma_{sk_F}(E_{pk_Q}(\mathcal{PS}_{F.Q})||exp)$, where $E(\cdot)$ denotes the ID-based encryption scheme. Note that the pseudonyms in the set will be arranged in a random order concatenated with each other, which is denoted as $||$. The "exp" represents the expiration time for the $\mathcal{PS}_{F.Q}$, in the sense that if the expiration time passes, the friends cannot use the original pseudonym set for establishing the anonymous communication. Since the users in the OSNs need to obtain friends' names for communication, we cannot hide users' IDs when they initiate the friendship establishment. To some extent, the reason that we implement the social approach for anonymous communication is to hide the identity and network address during the recommendation procedure, not in the initiation step. Thus, exposing the real identity in this stage does not impair users' privacy.

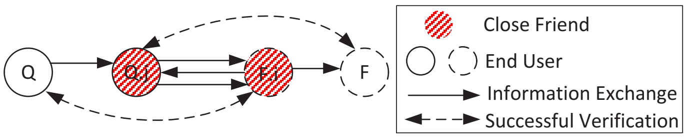


Fig. 3. Anonymous Close Friend Authentication

In the following, we present a pairing-based anonymous close friend authentication scheme as shown in Fig. 3. Here, we define a global hash function $\hat{H}(\cdot)$ which maps an arbitrary inputs to a fix-length output. The process is as follows:

1. $Q \to Q.j : E_{pk_{Q.j}}(PS_{F.Q}^{\alpha} || sk_{PS_{F.Q}^{\alpha}}), exp, \sigma_Q$
2. $Q.j \to PS_{F.i}^{\beta} : PS_{F.Q}^{\alpha}, n_1, \sigma_{Q.j}$
3. $PS_{F.i}^{\beta} \to Q.j : PS_{F.i}^{\beta}, n_2, \Phi_{\beta,\alpha} = \hat{H}(n_1||n_2||0||\mathcal{K}_{\beta,\alpha})$
4. $Q.j \to PS_{F.i}^{\beta} : \Phi_{\alpha,\beta} = \hat{H}(n_1||n_2||1||\mathcal{K}_{\alpha,\beta})$

Note $Q.j \in \mathcal{F}_Q^C$ is one of the close friends of Alice and σ is the corresponding ID-based signature. Then $Q.j$ uses $PS_{F.Q}^{\alpha}$ as its own pseudonym to authenticate himself to one of the pseudonyms that Bob gave to Alice during the previous step, e.g., $PS_{F.i}^{\beta}$ which is one of Bob's close friends. When the trusted user which behaves as $PS_{A.i}^{\beta}$ receives the packets, he/she will derive the session key as $\mathcal{K}_{\beta,\alpha} = e(H(PS_{F.Q}^{\alpha}), sk_{PS_{F.i}^{\beta}})$. Then $PS_{F.i}^{\beta}$ sends back the calculated $\Phi_{\beta,\alpha}$ which includes the session key. Upon receiving the packets, $Q.j$ derives the $\mathcal{K}_{\alpha,\beta}$ and checks whether $\Phi_{\beta,\alpha} = \hat{H}(n_1||n_2||0||\mathcal{K}_{\alpha,\beta})$, since we can determine the equation as follows,

$$\begin{aligned}\mathcal{K}_{\beta,\alpha} &= e(H(PS_{F.Q}^{\alpha}), \varsigma_F H(PS_{F.i}^{\beta})) \\ &= e(\varsigma_F H(PS_{F.Q}^{\alpha}), H(PS_{F.i}^{\beta})) = \mathcal{K}_{\alpha,\beta}.\end{aligned}$$

Accordingly, $Q.j$ knows that $PS_{F.i}^{\beta}$ is Bob's friend as well. In order to authenticate $Q.j$ to $PS_{F.i}^{\beta}$, $Q.j$ returns $\Phi_{\alpha,\beta}$ back. $PS_{F.i}^{\beta}$ can compute $\hat{H}(n_1||n_2||1||\mathcal{K}_{\beta,\alpha})$ and check whether it equals to the received packets which includes $\Phi_{\alpha,\beta}$. Therefore, two OSN users are able to mutually authenticate and securely communicate with each other.

4.3 Trust-based Friend Recommendation

The trust-based friend recommendation includes two major subprotocols, secure social coordinate matching and friend recommendation process. Based on the matching results (inner product) of social coordinates and established trust relationships, recommenders determine their recommendation decision on whether continue to query their friends or not.

4.3.1 Secure social coordinate matching

To achieve the secure social coordinate matching, we apply the secure kNN scheme in [41] and modify it as shown in Algorithm 1. In our scheme, users' social coordinates can be formed into a set of binary vector $\mathcal{A}$. Binary vector $\mathcal{Q}$ is the social coordinate vector that contains query information, which can be any possible user's unique social coordinate in the OSN. Note that $\mathcal{Q}[k] \in \{0,1\}$ has the same definition as $\mathcal{A}[k]$.

We define the degree of similarity as the inner product of the above two vectors, $\mathcal{P} = \mathcal{A} \cdot \mathcal{Q}$. At the beginning of Algorithm 1, CA selects a secret parameter $\mathcal{S}$ and two invertible matrices $\mathbf{B}_1, \mathbf{B}_2$ for each user as shown in Ln. 1-2. From Ln. 3-15, CA creates the extended vectors $\bar{\mathcal{A}}$ and $\bar{\mathcal{Q}}$ for the user's social attributes and the queried vector, and further embeds a random number r to secure the confidentiality of the matching results $\mathcal{P}$. Based on $\mathcal{S}$, $\mathbf{B}_1$, and $\mathbf{B}_2$, CA encrypts extended vectors as $\{\mathbf{B}_1^T \bar{\mathcal{A}}^{[1]}, \mathbf{B}_2^T \bar{\mathcal{A}}^{[2]}\}$ and $\bar{\mathcal{Q}}^{[2]}\}$ as $\{\mathbf{B}_1^{-1} \bar{\mathcal{Q}}^{[1]}, \mathbf{B}_2^{-1} \bar{\mathcal{Q}}^{[2]}\}$ from Ln. 16-24. The final matching result can be derived in Ln. 25.

Encrypted Social Coordinate Distribution: As an OSN user, he/she needs to obtain his/her 1-hop friends' social coordinates so that he/she can perform the above matched operation to derive the "best" matching friends. We ask CA to generate and distribute encrypted form of users' social coordinates to OSN users. We suppose both Bob(F) and Carol(R) are satisfied with the establishment of friendship, in the sense that Alice is able to query Bob for the recommendation on Carol in the future. Then, they mutually store each other's encrypted social coordinates. Assuming Bob is trying to add Carol to its friend list, the distribution process is as follows: (Carol also can follow the same procedure to add Bob as Carol's friend, respectively), where the $\Psi_{R.F}$ is a certificate that Carol gives to Bob, and it allows CA to issue the Carol's encrypted social coordinate to Bob. Note that Ψ will not reveal the specific value of trust level. Bob stores the encrypted social coordinates of all his 1-hops, which enables him to help recommend friends. The $Cert.Req$ and $Cert.Resp$ are the header of each of the packets to denote that their purposes are for certificate request and response, respectively. As we assumed before, all the communication between CA and each user of the system in secure channels are supposed to be uncompromisable. In addition, once there is an update for a particular user, all of his/her friends need to periodically update the encrypted social coordinate according to the expiration time, and this process can be done before the recommendation process initiates.

Query Initiation: Here, we consider two possible search patterns, social coordinate search and ID search, both of which can be done via current OSN service. First, we allow OSN users to search for a fuzzy identity without a specific ID, such as a user-defined social coordinate vector which represents the attributes of the user that they may look for, e.g., *a cardiologist, male, with more than 20 years work experience*. However, without involving the destination user's consent, Alice only needs to create the vector and sets the threshold that meets her desired social coordinates.

1. $R \rightarrow F : E_{pk_F}(\Psi_{R.F}), exp, \sigma_{sk_R}(E_{pk_F}(\Psi_{R.F})||exp)$
2. $F \rightarrow CA : Cert.Req, \Psi_{R.F}, exp$
3. $CA \rightarrow F : Cert.Resp, \mathbf{B}_{F1}^T \bar{\mathcal{A}}_R^{[1]}, \mathbf{B}_{F2}^T \bar{\mathcal{A}}_R^{[2]}$

Algorithm 1 Secure kNN Scheme

1: Randomly choose $(n+1)$ bit vector $\mathcal{S}$
2: Randomly choose $(n+1) \times (n+1)$ invertible matrices $\mathbf{B}_1, \mathbf{B}_2$
3: Extend column vector $\mathcal{A}$ to $(n+1)$ dimension
4: **if** $1 \leqslant k \leqslant n$ **then**
5: set $\bar{\mathcal{A}}[k] = \mathcal{A}[k]$
6: **else**
7: set $\bar{\mathcal{A}}[k] = -0.5||\mathcal{A}^2||$
8: **end if**
9: Extend column vector $\mathcal{Q}$ to $(n+1)$ dimension
10: **if** $1 \leqslant k \leqslant n$ **then**
11: set $\bar{\mathcal{Q}}[k] = \mathcal{Q}[k]$
12: **else**
13: set $\bar{\mathcal{Q}}[k] = 1$
14: **end if**
15: Extend $\mathcal{Q}$ to $(n+1)$ dimension $\bar{\mathcal{Q}}$, set the $(n+1)$ dimension as 1, $r > 0$, scale $\bar{\mathcal{Q}}$ as $(r\mathcal{Q}, r)$
16: **if** $\mathcal{S}[k] = 0$ **then**
17: set $\bar{\mathcal{A}}^{[1]}[k] = \bar{\mathcal{A}}[k]$, $\bar{\mathcal{A}}^{[2]}[k] = \bar{\mathcal{A}}[k]$
18: Random set $\bar{\mathcal{Q}}^{[1]}[k]$, $\bar{\mathcal{Q}}^{[2]}[k]$, let $\bar{\mathcal{Q}}^{[1]}[k] + \bar{\mathcal{Q}}^{[2]}[k] = \bar{\mathcal{Q}}[k]$
19: **else**
20: Random set $\bar{\mathcal{A}}^{[1]}[k]$, $\bar{\mathcal{A}}^{[2]}[k]$, let $\bar{\mathcal{A}}^{[1]}[k] + \bar{\mathcal{A}}^{[2]}[k] = \bar{\mathcal{A}}[k]$
21: set $\bar{\mathcal{Q}}^{[1]}[k] = \bar{\mathcal{Q}}[k]$, $\bar{\mathcal{Q}}^{[2]}[k] = \bar{\mathcal{Q}}[k]$
22: **end if**
23: Encrypt $\{\bar{\mathcal{A}}^{[1]}, \bar{\mathcal{A}}^{[2]}\}$ as $\{\mathbf{B}_1^T \bar{\mathcal{A}}^{[1]}, \mathbf{B}_2^T \bar{\mathcal{A}}^{[2]}\}$
24: Encrypt $\{\bar{\mathcal{Q}}^{[1]}, \bar{\mathcal{Q}}^{[2]}\}$ as $\{\mathbf{B}_1^{-1} \bar{\mathcal{Q}}^{[1]}, \mathbf{B}_2^{-1} \bar{\mathcal{Q}}^{[2]}\}$
25: **return** $\{\mathbf{B}_1^T \bar{\mathcal{A}}^{[1]}, \mathbf{B}_2^T \bar{\mathcal{A}}^{[2]}\} \cdot \{\mathbf{B}_1^{-1} \bar{\mathcal{Q}}^{[1]}, \mathbf{B}_2^{-1} \bar{\mathcal{Q}}^{[2]}\} = r\mathcal{A}\mathcal{Q} - 0.5r||\mathcal{A}^2||$

On the other hand, when Alice wants to find a particular OSN user, say David, among all users in the system, apart from knowing the real ID, Alice should know the encrypted social coordinate of David in order to let Alice's friends help her discover and recommend the trust chain. Similar to the above process, we do not allow Alice to obtain the plaintext of David's social coordinate. So, suppose David(D) agrees Alice(Q) to search himself,

1. $Q \rightarrow D : Cert.Req, E_{pk_D}(Q), t_1, \sigma_{sk_Q}(E_{pk_D}(Q)||t_1)$
2. $D \rightarrow Q : Cert.Resp,$
   $E_{pk_Q}(\mathcal{C}_{D.Q}), t_2, exp, \sigma_{sk_D}(E_{pk_Q}(\mathcal{C}_{D.Q})||t_2||exp)$

where t_1 and t_2 are timestamps in order to prevent replay attack. The $Cred.Req$ and $Cred.Resp$ denote the credential request and response, $\mathcal{C}_{D.Q}$ is the credential that David issues to Alice, meaning that David allows Alice to obtain David's encrypted social coordinate and search over the friends in the system and derive the corresponding trust level.

4.3.2 Trust-based Privacy-Preserving Friend Recommendation

Based on the intuition introduced in Sec. 2.1, we give the formal definition on our trust-based privacy-preserving recommendation process on the aspects of trust level and social coordinate matching results.

Definition 1: Given three users Alice(Q), Bob(F), and Carol(R), Alice and Carol are 1-hop friends of Bob, but they are strangers without the existing trust relationship. The trust level criteria of recommendation process for Bob is

$$T_{F.Q} \geqslant T_{F.R},$$

where $T_{F.Q}$ is Bob's trust level on Alice, and $T_{F.R}$ is his trust level on Carol, respectively. Note that the trust relationship between strangers depends on the 1-hop trust. To extend Definition 1, we require the following inequality is satisfied,

$$T_{R.F} \geqslant T_{R.D},$$

such that Carol is able to recommend to Alice her friend David after Bob recommends Carol to Alice.

Definition 2: Suppose Alice(Q) has a query vector $\mathcal{Q}$ and initiates the recommendation scheme, given $\mathcal{P}_\ell = \mathcal{A}_\ell \cdot \mathcal{Q}$ which denotes the ℓ-th recommender (includes 1-hop friend), the criteria on the matching results should satisfy,

$$\mathcal{P}_\ell = \begin{cases} \mathcal{P}_{\ell+1}, & \text{if } \mathcal{P}_{\ell+1} \geqslant \mathcal{P}_\ell \\ \text{abort}, & \text{otherwise} \end{cases}$$

in the sense that the inner product of recommended friends' social coordinates and the query vector should increase when the recommendation process extends hop-by-hop. Therefore, Alice continues to extend her trust chain only if the next recommender better match the current candidates. Otherwise, she will abort the process and initiate the process from her 1-hop friends.

Recommendation Process: The trust-based recommendation process should satisfy the above requirements, such that the trust chain could be set up according to the matching results and the trust requirement. For the completeness, we describe the whole algorithm pseudo-code in Algorithm 2. According to previous description, Alice is able to use the credential $\mathcal{C}_{D.Q}$ to obtain David's encrypted social coordinate. Besides, Alice should specify to CA that her 1-hop friends that she wants to query, so that CA can issue the corresponding encrypted social coordinate for the matching operation. Assuming the trust relationships among Alice, Bob, and Carol satisfy *Definition 1*, we give the privacy-preserving recommendation process as follows,

1. $Q \to CA : FQ.Req, \mathcal{C}_{D.S}, \Psi_{A.S}, exp, t_3$
2. $CA \to Q : FQ.Resp, \mathbf{B}_{Q1}^{-1}\bar{\mathcal{Q}}_D^{[1]}, \mathbf{B}_{Q2}^{-1}\bar{\mathcal{Q}}_D^{[2]}, t_4$
3. $Q \to F : FM.Req, \mathbf{B}_{Q1}^{-1}\bar{\mathcal{Q}}_D^{[1]}, \mathbf{B}_{Q2}^{-1}\bar{\mathcal{Q}}_D^{[2]}, t_5, \sigma_Q$
4. $F \to Q : FM.Resp, \mathcal{P}_1, E_{\mathbb{K}_F}(\Psi_{R.F}||exp),$ $E_{pk_Q}(PS_{R.F}^{\gamma}||pk_{R.F}^{\delta}/sk_{R.F}^{\delta}||\tau_{R.F}), t_6, \sigma_F$

Note that $\mathbb{K}$ is a symmetric key shared between CA and each OSN user, which intends to prevent queriers from learning each recommender's certificate. After CA decrypts the certificate Ψ encrypted by $\mathbb{K}$, it returns the corresponding encrypted social coordinates based on Ψ or rejects the query if it expires. Then, Alice sends the encrypted social coordinates to Bob. After Bob returns the best matching result on all his friends to Alice, she sends the query to her stranger Carol and repeats the same process until she reaches David.

Privacy Preservation Approach: To provide the identity and social coordinate privacy, we apply parts of the secure kNN computation scheme [41]–[43] and implement several modifications. We change the $(n+1)$-th entry of every user's social coordinate set $\bar{\mathcal{A}}$ to 1 instead of $-0.5||\mathcal{A}^2||$ during the dimension extension. The extended dimension of each query vector $\mathcal{Q}$ is changed to $\bar{\mathcal{Q}} = (rQ, \bar{t})$, where $\bar{t}$ is a random number selected by OSN users, and they need to report $\bar{t}$ to

Algorithm 2 Trust-based privacy-preserving friend recommendation (pseudo-code)

for $i = 1 \to \max\{hop\}$ **do**
  if $\mathcal{P}_{i+1} < \mathcal{P}_i$ **then**
    abort;
  else
    $Q \to R_i(F)$: Matching request;
    $R_i(F) \to Q$: Return encrypted $\Psi_{R_i(F).Q}$;
    $Q \to CA$: Certificate $\mathcal{C}_{D.S}$, encrypted $\Psi_{R_i(F).Q}$;
    $CA \to Q$: Social Coordinate $\mathbf{B}_{R_i1}^{-1}\bar{\mathcal{Q}}_D^{[1]}$, $\mathbf{B}_{R_i2}^{-1}\bar{\mathcal{Q}}_D^{[2]}$;
    $Q \to R_i(F)$: Commitment $\tau_{R_i.Q}, \mathbf{B}_{R_i1}^{-1}\bar{\mathcal{Q}}_D^{[1]}$, $\mathbf{B}_{R_i2}^{-1}\bar{\mathcal{Q}}_D^{[2]}$;
    for $j = 1 \to |\mathcal{F}_{R_i(F)}|$ **do**
      $M = \{\mathbf{B}_{R_i1}^T\bar{\mathcal{A}}_{R_i.j}^{[1]}, \mathbf{B}_{R_i2}^T\bar{\mathcal{A}}_{R_i.j}^{[2]}\}\cdot\{\mathbf{B}_{R_i1}^{-1}\bar{\mathcal{Q}}_D^{[1]}, \mathbf{B}_{R_i2}^{-1}\bar{\mathcal{Q}}_D^{[2]}\}$
      if $T_{R_i.R_{i+1}} < T_{R_i.R_{i-1}}$ and $M_i < M_{i+1}$ **then**
        Choose $\max\{M\}$ and derive $\mathcal{P}_i$;
        Return R_{i+1} as next recommender;
      else
        Choose another R_i with lower M;
      end if
    end for
    $R_i \to S : R_{i+1}$, pk/sk key pair, Commitment $\tau_{R_{i+1}.R_i}$.
  end if
end for

CA before the encrypted social coordinate distribution. After Bob receives Alice's query, Bob first verifies the authenticity of the query vector. If the vector cannot be verified, he aborts the algorithm; otherwise, Bob checks all of his friends' encrypted social coordinates stored in its local storage to compute the inner product of two vectors as follows,

$$\begin{aligned} M &= \{\mathbf{B}_{F1}^T\bar{\mathcal{A}}_R^{[1]}, \mathbf{B}_{F2}^T\bar{\mathcal{A}}_R^{[2]}\}\cdot\{\mathbf{B}_{F1}^{-1}\bar{\mathcal{Q}}_D^{[1]}, \mathbf{B}_{F2}^{-1}\bar{\mathcal{Q}}_D^{[2]}\} \\ &= \bar{\mathcal{A}}_R^{[1]}\cdot\bar{\mathcal{Q}}_D^{[1]} + \bar{\mathcal{A}}_R^{[2]}\cdot\bar{\mathcal{Q}}_D^{[2]} \\ &= \bar{\mathcal{A}}_R\cdot\bar{\mathcal{Q}}_D = r(\mathcal{A}_R\cdot\mathcal{Q}_D) + \bar{t}. \end{aligned}$$

Here, we use $\mathcal{A}_R$ to denote the social coordinates of all the possible recommenders within 1-hop friendships with Bob, e.g., Carol. Then, Bob ranks all of the matching results of M according to the linearity on both r and $\bar{t}$. However, based on the trust levels that Bob gives on his friends, he only returns the candidates both have the $\max\{M\}$ and satisfy the trust level requirement in *Definition 1*. Then, with the knowledge of r and $\bar{t}$, Alice is able to derive $\mathcal{P}_1$ based on the returned M. If the results satisfy *Definition 2*, Alice repeats the same process to query Carol and further recommenders until discovering the destination user by observing the matching results. Finally, if all of the social coordinates are matched, the destination user, David, is reached based on the anonymous trust chain.

4.4 Trust Level Derivation

4.4.1 Design Objective

The basic requirement of trust level derivation process is securely collect the overall trust level based on each individual's value on the trust chain. According to the the assumptions in the previous section, OSN users treat their trust levels on the friends as privacy and do not want to disclose. To solve this dilemma, we apply part of the scheme in [44] to derive an overall value without compromising each user's private data. Although there are numerous works discussing how to derive the overall trust level based on each individual value, few of them considers the problem of securely collecting without revealing each value. In this work, we give a possible solution on securely collecting and deriving the average trust level [16], [34] on a particular trust chain.

4.4.2 Basic Construction

In the previous recommendation process, to confirm the receipt of the next hop's information and corresponding token, Alice sends an acknowledgement packet back to each of the recommended strangers. Here, we extend the format of the original ACK packet into $< ACK, sid, \epsilon_{ID}^{Q}, exp >$, where ACK is the header of the packet, $sid \in Z$ is an sequence number to guarantee the correctness of packet delivery, and ϵ_{ID}^{Q} is a commitment used to certify the next hop recommender. For example, Bob verifies the ACK packet from Alice and store ϵ_{F}^{Q} for a record, in the sense that Bob knows that the record corresponds to the next hop OSN user, Carol. When an OSN user sends Bob a packet with this record, Bob sends back the trust level back to him/her with privacy-preserving approach. Note that ID should be pseudonyms used for anonymous communication introduced in previous subsection.

In what follows, we provide a solution to derive the average trust level on the trust chain to represent the end-to-end trust level.

Setup: Alice asks CA to choose and publish a random public generator $\hat{g} \in G_1$ and $m+1$ random secrets $x_0, x_1, ..., x_m \in Z_q$ according to the number of hops, m, where $\sum_{i=0}^{m} x_i = 0$. Then, CA encrypts x_i based on the pseudonyms provided by Alice, such that only OSN users who have been given the designated pseudonyms can obtain the secret numbers. Here, we use $Agg.Req$ and $Agg.Resp$ to distinguish the packets,

1. $Q \rightarrow CA : Agg.Req, \{\epsilon_{R_i}^{Q}\}, \{R_i\}, HMAC_{\mathbb{K}_Q}(\epsilon_{R_i}^{Q}||R_i)$
2. $CA \rightarrow R_i : Agg.Resp, x_i, \epsilon_{R_i}^{Q}, HMAC_{\mathbb{K}_{R_i}}(x_i||\epsilon_{R_i}^{Q})$

where $0 \leqslant i \leqslant m-1$, and we use R_0 to represent Alice's 1-hop friend (F). Note that HMAC is hash-based message authentication code. We also refer R_i as the recommenders' pseudonyms that the querier used to communicate during the recommendation process. Similar to other recommenders, Alice is given the secret number x_0.

Encryption: After receiving the encrypted secret number, recommenders can encrypt their trust level if they can verify the authenticity of the packets,

$$R_i \rightarrow Q : \\ E_{pk_Q}(\hat{g}^{T_{i+1}} H_0(sid)^{x_i}), \sigma_{sk_{R_i}}(E_{pk_Q}(\hat{g}^{T_{i+1}} H_0(sid)^{x_i})),$$

where the cryptographic hash function H_0 is defined as a map $H_0 : Z \rightarrow G_1$, and T_i is the trust level from R_{i-1} to R_i (we refer R_{-1} as the querier). Based on the commitment $\epsilon_{R_i}^{Q}$, recommenders are able to find out the next recommender that they send to the querier, such that they can locate their records and generate the encrypted trust level.

Aggregation: Then, Alice collects all the results coming from R_i and derives the average value in the following way,

$$V_a = H_0(sid)^{x_0} \cdot \hat{g}^{T_0} \cdot \prod_{i=1}^{m} \hat{g}^{T_i} \cdot H_0(sid)^{x_i} = \hat{g}^{\sum_{i=0}^{m} T_i},$$

where T_0 is the trust level from the querier to her 1-hop friend.

Decryption: To decrypt the sum of $\sum_{i=0}^{m} T_i$, it suffices to compute the discrete log of V base with $\hat{g}$. Since our plaintext space is relatively small (we can define how fine-grained the trust level would be), decryption can be achieved through a brute-force method. We will give an efficiency analysis in Section V. A better approach would be the Pollard's lambda method [45] which requires time roughly square root of the plaintext space.

Then, Alice can derive the average trust level as $\bar{T} = \sum_{i=0}^{m} T_i/(m+1)$, which shows an average trust level on the trust chain. Given this approach, we can modify it into a more complex transitive trust metric defined in the literature or consider it in a multi-path scenario, where OSN users can be reached via different trust chains. We can assign parameters or weights on different paths to achieve more reasonable results on deriving end-to-end trust level.

5 Theoretical Evaluation

5.1 Security Analysis

In this subsection, we conduct the security analysis on our proposed scheme and discuss the possible attacks that our scheme is able to defend against in each step of the scheme.

5.1.1 Attacks on Friendship Establishment

The identity and network address tracing attacks severely deteriorate the user privacy and system reliability. **Type I** and **Type II** adversaries may can collect OSN users' pseudonyms in order to trace the real identity and the network address.

Type I adversaries may attempt to uncover the real identity of a particular OSN user. Our scheme enables OSN users to establish anonymous trust chain with their strangers, where users are assigned a set of collision-resistant pseudonyms to realize anonymous communications. Active adversaries can observe all the behaviors of a pseudonym, but OSN users involved with in the recommendation scheme will frequently change their pseudonyms, which provides the privacy of their real identities.

Another possible tracing attack can be considered as address attack, where both the MAC and IP address can become the targets of **Type II adversaries**. We suppose every OSN user in our system is assigned an unique MAC address and a variable IP address. Fortunately, our scheme is survived from this kind of attack. Note that our end-to-end communication is based on the relay of trusted users. Therefore, hidden by trusted users, the communication only exposes the trusted friends' MAC addresses instead of their real MAC addresses, which means adversaries cannot trace the interacted users by eavesdropping their MAC addresses. Similarly, we implement the sufficient large sets of pseudonyms for securing the anonymous communication, where the address of the pseudonym helps the real end user hide from disclosing the real IP address. Furthermore, analyzing the IP addresses of trusted users will not help locate the real IP address of end user, since the IP addresses of the friends are independent in the online social networks, e.g., everyone can have friends all around the world. To the contrary, revealing the IDs of friends will effectively enhance the possibility of tracing back the end user, where we use pseudonyms to prevent the ID from being traced.

5.1.2 Attacks on Trust-based Recommendation

In this process, we mainly discuss the possible attacks on the social coordinate. **Type IV adversaries** may intend to change their social coordinates in order to obtain other's social attributes, which mostly happens if adversary performs as a querier and initiates multiple queries for recommendation. For example, Alice can change the values of two vectors $\mathbf{B}_{F1}^{-1}\bar{\mathcal{Q}}_{D}^{[1]}$ and $\mathbf{B}_{F2}^{-1}\bar{\mathcal{Q}}_{D}^{[2]}$, both of which are directly obtained from CA. Although they have been encrypted with unknown parameters, the adversary is able to change the value of each element in the two vectors, which may result in abnormal inner product results. However, for a large dimensional vector, it is still infeasible for the adversary to derive others' social coordinates, due to the fact that the original vector has been changed if the adversary changes the encrypted vectors, in the sense, he/she cannot tell the difference according to what he/she has changed. On the other hand, **Type IV adversary** may be recommenders, where they intend to change their 1-hop friends' encrypted social coordinates with the purpose of discovering the querier's social coordinates. For the same reason, adversaries may fail to find out the true vectors.

One of the design goals is to provide the identity privacy of the querier, because, for example, the behavior of requiring a recommendation of a doctor may potentially leak her privacy. Since we apply the encrypted social coordinate vectors to query, the privacy of queried information can be preserved. For the **Type IV adversaries** who intend to obtain the information regarding the encrypted social coordinates, they can only perform the matching operations among their 1-hop friends and derive the corresponding results. Although they are able to rank all the results, they cannot find the matching detail without the value of r and $\bar{t}$, e.g., which one of the attributes are the same. Moreover, due to the insufficient knowledge of their 1-hop friends, it also prevents them from knowing queried information. Therefore, we preserve the privacy of the querier in terms of what she queries and her identity.

5.1.3 Attacks on Trust Level Derivation

During the trust level derivation process the **Type I adversary** will launch active attacks like bogus data injection and passive attacks during the packet delivery. By implementing the ID-based signature scheme on every packet, adding new values or maliciously replacing the values will not help enhance or decrease the existing trust level. Because every recommender issues the signature based on the identity, the querier will not accept the result if the verification fails. For the malicious querier, he/she may want to compromise each recommender's trust level on the friend chain. However, no user is able to derive the T_i from $\hat{g}^{T_i} \cdot H_0(sid)^{x_i}$ due to the assumption that Decisional Diffie-Hellman is hard. Note that although we can utilize brute force or Pollard lambda method to derive $\sum T_i$ from $\hat{g}^{\sum T_i}$, we cannot implement the same approaches to obtain T_i from $\hat{g}^{T_i} \cdot H(sid)^{x_i}$ due to the plaintext space is much greater than $\sum T_i$ and the unknown secret x_i.

Another type of attack launched by the **Type I adversary** is by requesting CA to generate multiple sets of secret numbers on one trust chain to obtain private trust levels. For example, the adversary first requests the actual number of hops as 6, and fraudulent requests 5 for the second time. Then, by comparing different results with different number of hops, he can discover the trust level between the recommenders that have been excluded during the second request. However, our scheme is able to defend this attack by our anonymous authentication scheme. Since most recommenders to the querier are strangers, they use frequently changed pseudonyms during each process. What the querier obtains during the recommendation process is a pseudonym that each recommender assigns to his/her close friends. Therefore, without a clear match between real identities and trust relationships, the trust level cannot be leaked only based on path information.

Type III adversaries, performed as malicious recommenders, may compromise the trust level on each trust chain. During the trust derivation process, they impersonate as good OSN users and want to obtain trust level on a particular trust chain that they do not belong to. To overcome this attack, our scheme requires the same sid during the derivation process. Without sufficient knowledge of both sid and the corresponding secret number x_i, the querier cannot derive the correct end-to-end trust level. Since the querier has the recommender records during the recommendation process, the adversary can be identified during the signature verification process.

5.2 Complexity Analysis

5.2.1 Storage Cost Analysis

In our simulation settings, we use the degree of the curve as 2, which gives the element of size 512-bit in both G_1 and G_2. For each OSN user, to store their assigned pseudonyms and key pairs costs $2\kappa|G_1|$. The encrypted social coordinates of each friend may cost 160 bits, and the total storage cost for storing OSN users' friends depends on the number of their friends. Besides, the commitment of trust level for each OSN users' friend costs $|G_1|$. Based on the observation over Facebook [46], the total cost for storing the above parameters is less than 300KB given the fact that the average number of friends is 150 and $\kappa = 200$. We also consider the storage cost for CA. In our scheme, the key for each OSN user's social coordinate consists of an $O(n) \times O(n)$ matrix and an $O(n)$ vector. Assuming there are $|\mathcal{V}|$ OSN users in the system, to store all the social coordinates costs $O(|\mathcal{V}|n)$ storage resources. In addition to the social coordinates, CA also needs to store OSN users' IDs and the corresponding pseudonyms, which costs $O(|\mathcal{V}|)$. Thus, the total storage of CA would be $O(|\mathcal{V}|n + n^2)$, where n is a tunable parameter depending the security level of the system.

5.2.2 Communication Cost Analysis

First, we consider the communication cost before the recommendation process. It requires $O(1)$ between CA and each OSN user, while $O(N)$ for storing friends' encrypted social coordinates, where N denotes the number of friends of each OSN user. Second, during the recommendation process, the querier needs to communicate with CA for $O(\ell)$ times. For the recommendation on the trust chain, it requires $O(\ell N)$ to discover the destination users. In the trust level derivation process, it requires $O(1)$ between recommenders and CA, and $O(\ell)$ between recommenders and the querier.

6 Performance Evaluation

6.1 Experimental Evaluation

To evaluate the performance of our scheme, we use the Facebook dataset [47] and INFOCOM 2006 dataset [48] to analyze the proposed scheme in terms of routing performance. Based on the scheme description, we may consider our scheme as a routing protocol among OSN users, where the routing metric jointly considers the trust relationships and social coordinate matching results. Although the INFOCOM dataset that we use is not a real OSN, we assume that attendees form an OSN after their frequent social interactions during the conference. We also highlight the *Number of Average friends* in Table 2 and *Number of Average Contact Users* in Table 3, both of

TABLE 2
Facebook Dataset

University Name	Caltech	Reed	Haverford
Number of Users	762	962	1,446
Length of the Experiment	1 day	1 day	1 day
Number of Existing Friendship	33,302	37,624	119,178
Number of Possible Friendship	579,882	924,482	2,089,470
Number of Average Friends	21.9	19.5	41.2
Social attributes used / Total	7 / 7	7 / 7	7 / 7

TABLE 3
INFOCOM 2006 Dataset

Number of Users	78
Length of the Experiment	4 days
Contact Detection Period	120 sec
Average Contact Duration	511.4 sec
Number of Average Contact User	63.7
Social attributes used / Total	11 / 17

In the experimental evaluation, we mainly focus on analyzing the reachability between two arbitrary users for the datasets, where the reachability is defined as follows,

$$\mathcal{R}_i = \frac{\sum_j \mathcal{E}_{ij}}{\mathcal{E}_{total}},$$

where $\mathcal{E}_{ij}$ is the j-th trust chain between two strangers involving i recommenders ($i+1$ hops to reach the destination user), and $\mathcal{E}_{total}$ is the total number of possible connections in the network. In addition, since the INFOCOM 2006 dataset contains contact duration information, we further utilize this to evaluate its impact to the reachability.

For our experiment settings, we use different length of bits to represent the attribute values, e.g., *gender: male*:01, *gender: female*:10, or *nationality: US*: 000001, *nationality: China*:100000, where the number of possible values is the length of the social coordinates. We further use these attribute sets to represent each OSN user in the experiment. For the existing relationship and possible relationship, we consider it as asymmetric pairwise relationships.

6.1.1 Exprimental Results

- **Facebook Dataset**

First of all, we carry out the analysis on the reachability of our proposed scheme based on the collected Facebook data from three universities, California Institute of Technology, Reed College, and Haverford College. As shown in Fig. 4, our scheme greatly increases the reachability between two arbitrary users on Facebook, from 5.74% to 81.56%, 4.07% to 84.54%, and 5.70% to 89.19%, respectively. The result also indicates that the multi-hop trust chains between two arbitrary users could be established via the progressive matching results on users' identical attributes. Note that we consider the asymmetric friendship chain in our experiment.

Among all the trust chain established between OSN users, we investigate the distribution of the number of recommenders

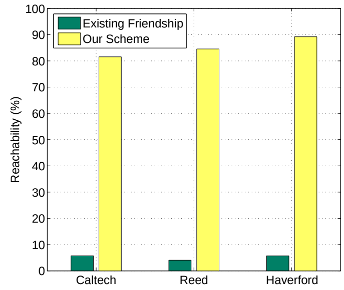


Fig. 4. Reachability Comparison between Existing Friendship and Our Scheme

on each trust chain in the OSN. As shown in Fig. 5, most of the newly established trust chains require less than 3 hops for completing the recommendation process, which are 75.9%, 71.2%, and 80.8% for Caltech, Reed, and Haverford, respectively. Particularly, we want to point out that the numbers of ID-based recommendation within 2 hops are 340,332, 477,062, and 1,406,254, which are greater than our scheme 212,664, 250,392, and 783,589. The reason for that is our scheme will first filter out "unqualified" recommenders, and only forward to friends whose number of identical attributes with the destination user is greater than the current matching result. In addition, the decision on progressive matching results requires $\mathcal{P}_{\ell+1} \geqslant \mathcal{P}_{\ell}$, which indicates the possibility of the equality of identical attributes on two or more consecutive hops. We can see from Fig. 5, although the compared number of attributes is 7 in the Facebook dataset, but we may have multi-hop trust chains including more than 8 recommenders. Hence, we increase the possibility of reaching more OSN users that are more than 7 hops away.

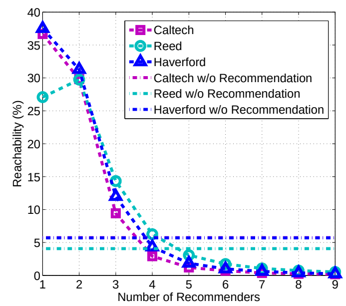


Fig. 5. Reachability Distribution against the Number of Recommender

- **INFOCOM 2006 Dataset**

To better evaluate the important role of attributes, we take a step further to investigate the INFOCOM dataset which contains more than 17 attribute information. Different from the Facebook dataset, the users in the dataset are closely connected according to the contacts. As we can compare from Table 2 and Table 3, the degree of each user in INFOCOM dataset is 63.7 (after removal of incorrect records), which are larger than the numbers in Facebook dataset. However, some of the contacts either are incorrectly recorded or cannot reflect real physical contacts. Therefore, we have to evaluate the impact of the contact duration and the reachability. The general contact duration for each interaction is shown in Fig. 6, and there are more than 140,000 contacts collected by

Bluetooth devices. However, most of the contact durations are less than 2 min, where we consider that the involved users may not have real interactions, rather, they may just stay in the transmission range of their Bluetooth devices and leave the incorrect records.

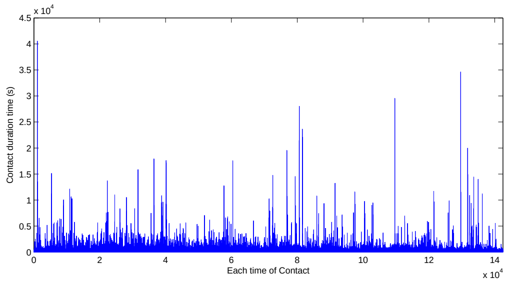


Fig. 6. Contact durations in INFOCOM 2006.

In the experimental analysis on INFOCOM dataset, we raise a generally-accepted hypothesis on evaluating the trust relationships in our scheme, where more number of contacts (above certain level of durations) means more trustful [49]. Accordingly, if a pair of users frequently contacts with each other, we may consider they mutually trust each other compared to other users. The following experiment results compare the different routing performance given progressive duration of contact duration (from 0 min to 10 min) and given trust levels.

We first evaluate the reachability in terms of establishing trust chain between two OSN users via the multi-hop recommendation process. Based on the observation in Fig. 7, the 1-hop reachability decreases dramatically when the contact duration is set larger, which has the similar results as in Fig.4. For the case that contact duration is set to 0, the multi-hop

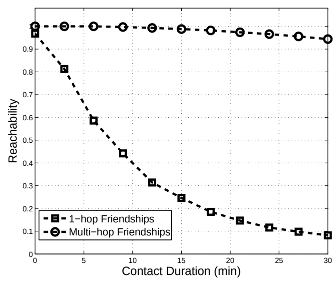


Fig. 7. Performance Evaluation in INFOCOM 2006.

reachability is as low as 3.5% in Fig. 8(a), which shows most of the social relationships are formed using 1-hop trust relationship. For the same reason, we can only find less than 4 multi-hop trust chains with at least 3 recommenders. However, with the increment of contact durations, which indicates only longer interactions are taken into consideration, the multi-hop social relationships become the major reason that forms the end-to-end trust relationship. As an example, the reachability between two strangers increases from 0.08% to 17% on a three-hop trust chain as shown in Fig.8(a) and Fig.8(d). As shown in Fig. 8(c), the number of maximum recommenders on a trust chain achieves to 7 with the consideration on trust level, while arbitrary two strangers are able to connect with each other within 4-hop without each other's requirement on trust level. We can clearly see the increase on the number of recommenders from Fig. 8, where the number of hops is 6 if we remove the criterion on *Definition 1*, while OSN users create a 9-hop trust chain if we apply trust levels in recommending strangers. Compared with the performance of Facebook dataset, they share similar decrease trend in terms of the reachability ratio, but the number of hops and the corresponding number of possible connections in the Facebook dataset may be larger than that in the INFOCOM dataset, because Facebook dataset involves more users and possible connections.

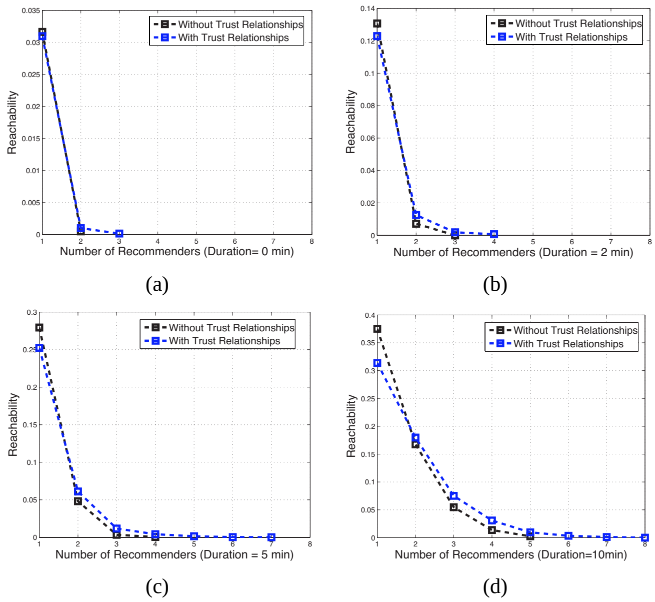


Fig. 8. Reachability of the Proposed Scheme VS Contact Duration

The above observation verifies our motivation on designing the trust-based recommendation scheme, where the trust relationship can be used to establish multi-hop relationship, but the subjective trust level would lower its possibility and further extent the number of hops.

6.1.2 Comparison with Other Schemes

To further evaluate the performance of our scheme, we use the Facebook dataset to compare the reachability of our scheme and other recommendation schemes. In this experiment setting, we calculate the accumulated number of connections between two arbitrary users in the network. We mainly compare our scheme with non-recommendation performance (as the baseline), ID-based recommendation approach [7], and Talash approach in [8]. As shown in Fig. 9, since traditional ID-based recommendation schemes lack of ability of extending recommendation chain, it has the lowest reachability as 41.65%, 31.15%, and 43.21% in three datasets. Talash approach achieves better performance due to their analysis on social attributes, which has the same design intuition with our work. However, they did not discuss the possibility of multi-hop chains that are 2 hops, which becomes the main reason that causes the lower reachability than our scheme. For our scheme, if we take 1-hop recommendation as successful trust chain establishment, the accumulative reachability ratio will be 85.71%, 88.62%, and 94.90% for Caltech, Reed, and Haverford, respectively.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TDSC.2014.2355824

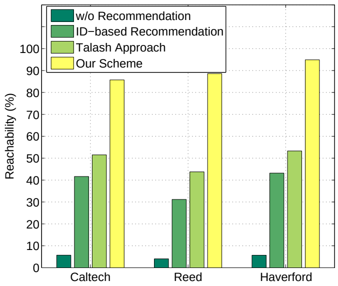
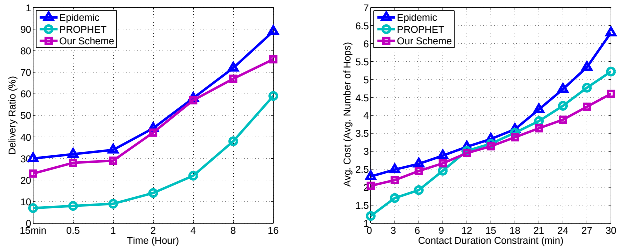


Fig. 9. Comparison with Other Recommendation Schemes



(a) Reachability

(b) Avg. Number of Hops

Fig. 10. Comparison with Other Packet Forwarding Schemes

We also compare our recommendation scheme with some packet forwarding schemes in social networks in order to analyze the performance against time constraint. We consider our approach is comparable with these schemes in terms of choosing best relay users to improve reachability and reduce cost. In the following experiment, we choose two well known approaches, epidemic routing [50] and PROPHET [51], by using the INFOCOM 2006 dataset as our scheme. In Fig. 10(a), we investigate the reachability changes in corresponding with the time. The epidemic approach has the best reachability, and it reaches to nearly 89% when the experiment lasts for more than 16 hours. The reason for that is users will automatically exchange information when they contact each other. Although it has the best reachability performance, this approach brings a lot of network traffic burden. Our approach is obvious better than the PROPHET approach from the beginning of the experiment, where the reachability our scheme is close to 77% by the end of simulation. We also try to explore the efficiency of our proposed scheme on the aspect of number of hops. For a recommendation scheme, less number of hops indicates that queriers would be easier to establish a multi-hop trust chain. As shown in Fig. 10(b), the average number of hops of our scheme will reach to 4.5 for multi-hop trust chain when the time duration is set to 30min, while other two schemes have more cost on number of recommenders compared to ours. Generally speaking, in terms of cost, our scheme outperforms the other two schemes when the contact duration is set less than 12min. For the delivery ratio, our scheme is better than PROPHET. In corresponding to the results in Fig. 7, we use multi-hop trust chain and attribute matching approach to compensate the deficiency the reachability given by 1-hop friendship, and efficiently achieve better reachability in terms of number of recommenders.

6.2 Efficiency Analysis

We will discuss the computational cost of our scheme in different stages. We use Pairing-based Cryptography (0.5.12) Library to implement our simulation. We take Tate pairing as our basic pairing operation. The elliptic curve we use for our scheme is type A. A curve of such type has the form of $y^2 = x^3 + x$. To achieve the 80-bit security level (same as 1024-bit RSA), the order of the curve is around 160 bits, and the base field is F_p where $|p| = 512$. For the experiments, we use a laptop with an Intel processor 2.8GHz and 1GB RAM under the platform Ubuntu 11.10. All the timing reported are averaged over 100 randomized runs.

6.2.1 *Privacy-preserving friendship establishment.*

The major computational cost for OSN users during this process is the authentication process. As in [36], our encryption scheme will incur one pairing operations, one scalar multiplication in G_1 and one exponentiation operation over G_2. The decryption process yields one pairing computation as well. According to [52], the signing process for generating ID-based signature costs one exponentiation in G_2, and one multiplication in G_1, while the verification incurs one exponentiation operation in G_2 and two pairing operations. The verification in securing the anonymous communication will incur one encryption and one signing. For the trusted user, it has the burden of one pairing on decryption and one on verification. Both of the users have to derive the session key, which costs 1 pairing for each. Therefore, to establish the anonymous communication between two OSN users will cost 5 pairing operations and 3 exponentiations on G_2 for each one. Based on our results, the exponentiation operation takes 5.3ms, while a pairing operation takes 15.2ms.

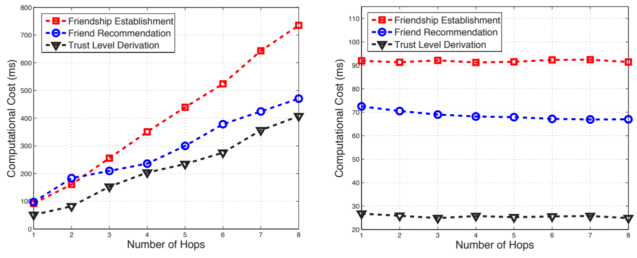
6.2.2 *Friend recommendation process.*

For the recommendation process, the querier consumes $2\ell+2$ exponentiation operations over G_2 and $3\ell + 2$ pairing operations, where ℓ is the number of recommenders. As each recommender, they take 3 exponentiation and 3 pairing operations during this process apart from the matching computation over encrypted social coordinates. Since we require the 80-bit security level (same as 1024-bit RSA), we set $n \geqslant 80$ in order to defend the attack which tries to compromise the encryption scheme on the social coordinates.

6.2.3 *Trust level derivation.*

This process incurs 1 pairing and 2 exponentiation operations for each recommender on the trust chain, while the querier consumes ℓ exponentiations and 3ℓ pairing operations. For implementing the brute-force in deriving the average trust level from $\hat{g}^{\sum T_i}$, according to [44], it takes only 0.3ms to compute a modular exponentiation using high-speed elliptic curve as $curve25519$. The encryption part for the trust level takes 0.6ms. Thus, decryption requires a discrete log which takes approximately 0.3ms to try each possible plaintext. Based on our simulation settings, we have the following results in Fig. 11, where we consider the longest trust chain shown in Fig. 8(d).

As we can see from Fig. 11(a), the computational costs in three phases grow nearly linearly when the number of hops increases, since the querier mostly repeats the recommendation



(a) Querier (b) Recommender

Fig. 11. Computational Cost of the Proposed Scheme.

process until he/she finds the destination user. Especially, the trust level derivation process consists of two computational parts, one is decrypting the packets from recommenders, while the other one is decrypting the end-to-end trust level. However, comparing to the former part, the computational cost of decrypting the final results consumes negligible time. Thus, the total computational cost in this stage is mainly on decrypting recommenders' ciphertexts. In general, even if the trust chain is set up to 9 hops, the total costs for the querier is less than 2s. For each recommender shown in Fig. 11(b), the computational costs in the friendship establishment and trust level derivation stages remain flat, where the most consuming part takes less than 90ms. Particularly, one feature in analyzing the simulation results is that the computational cost in recommendation phase slightly drops when the number of hops increases. It may result from the fact that the number of recommenders' friends reduces, such that the time in deriving the matching results may decrease. Therefore, based on the above analysis, we show the efficiency of each OSN user in our proposed scheme.

7 Conclusion

In this paper, we propose a privacy-preserving trust-based friend recommendation scheme for online social networks, which enable two strangers establish trust relationships based on the existing 1-hop friendships. For privacy concerns, we first design the anonymous close friend authentication scheme to secure the communication among OSN users. Then, we apply the secure kNN computation as the running protocol to derive the encrypted social coordinate matching results. To derive the objective trust level, we propose a solution to calculate the average trust level as the transitive overall value without compromising each individual's trust level. Through security analysis and experimental evaluation, we have shown the security and feasibility of the proposed scheme.

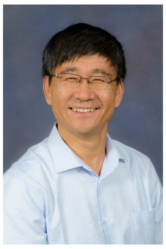
References

[1] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 2007, pp. 29–42.
[2] C. Zhang, X. Zhu, Y. Song, and Y. Fang, "A formal study of trust-based routing in wireless ad hoc networks," *INFOCOM 2010. 29th IEEE International Conference on Computer Communications. IEEE*, pp. 1 – 9, Mar. 2010.
[3] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*. IEEE, 2008, pp. 506–515.
[4] T. H.-J. Kim, A. Yamada, V. Gligor, J. Hong, and A. Perrig, "Relationgram: Tie-strength visualization for user-controlled online identity authentication," in *Financial Cryptography and Data Security 2013*, 2013.
[5] L. Backstrom, E. Sun, and C. Marlow, "Find me if you can: improving geographical prediction with social and spatial proximity," in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10, 2010, pp. 61–70.
[6] R. Dey, C. Tang, K. Ross, and N. Saxena, "Estimating age privacy leakage in online social networks," in *INFOCOM, 2012 Proceedings IEEE*, 2012, pp. 2836–2840.
[7] M. von Arb, M. Bader, M. Kuhn, and R. Wattenhofer, "Veneta: Serverless friend-of-friend detection in mobile social networking," in *Networking and Communications, 2008. WIMOB '08. IEEE International Conference on Wireless and Mobile Computing,*, oct. 2008, pp. 184 – 189.
[8] R. Dhekane and B. Vibber, "Talash: Friend finding in federated social networks." in *LDOW*, 2011.
[9] C. Dwyer, S. R. Hiltz, and K. Passerini, "Trust and privacy concern within social networking sites: A comparison of facebook and myspace." in *AMCIS*, 2007, p. 339.
[10] C. Zhang, J. Sun, X. Zhu, and Y. Fang, "Privacy and security for online social networks: challenges and opportunities," *Network, IEEE*, vol. 24, no. 4, pp. 13–18, 2010.
[11] P. W. L. Fong, M. Anwar, and Z. Zhao, "A privacy preservation model for facebook-style social network systems," in *Proceedings of the 14th European Conference on Research in Computer Security*, ser. ESORICS'09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 303–320.
[12] B. Carminati, E. Ferrari, and A. Perego, "Enforcing access control in web-based social networks," *ACM Trans. Inf. Syst. Secur.*, vol. 13, no. 1, pp. 6:1–6:38, Nov. 2009.
[13] A. C. Squicciarini, M. Shehab, and F. Paci, "Collective privacy management in social networks," in *Proceedings of the 18th International Conference on World Wide Web*, ser. WWW '09. New York, NY, USA: ACM, 2009, pp. 521–530.
[14] A. Squicciarini, F. Paci, and S. Sundareswaran, "Prima: a comprehensive approach to privacy protection in social network sites," *Annals of telecommunications*, vol. 69, no. 1-2, pp. 21–36, 2014.
[15] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: inferring user profiles in online social networks," in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 251–260.
[16] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decis. Support Syst.*, vol. 43, pp. 618–644, March 2007.
[17] J.-H. Cho, A. Swami, and I.-R. Chen, "A survey on trust management for mobile ad hoc networks," *Communications Surveys Tutorials, IEEE*, vol. 13, no. 4, pp. 562–583, 2011.
[18] K. Govindan and P. Mohapatra, "Trust computations and trust dynamics in mobile adhoc networks: A survey," *Communications Surveys Tutorials, IEEE*, vol. 14, no. 2, pp. 279 –298, quarter 2012.
[19] W. Sherchan, S. Nepal, and C. Paris, "A survey of trust in social networks," *ACM Comput. Surv.*, vol. 45, no. 4, pp. 47:1–47:33, Aug. 2013.
[20] R. Guha, R.Kumar, P. Raghavan, and A. Tomkins, "Propagation of trust and distrust," *Proceedings of the 13th international conference on World Wide Web*, pp. 403–412, 2004.
[21] L. Guo, X. Zhu, C. Zhang, and Y. Fang, "A multi-hop privacy-preserving reputation scheme in online social networks," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, Dec 2011, pp. 1–5.
[22] P. Lin, P.-C. Chung, and Y. Fang, "P2p-isn: a peer-to-peer architecture for heterogeneous social networks," *Network, IEEE*, vol. 28, no. 1, pp. 56–64, 2014.
[23] E. Daly and M. Haahr, "Social network analysis for information flow in disconnected delay-tolerant manets," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 5, pp. 606 –621, may 2009.
[24] W. Chen and S. Fong, "Social network collaborative filtering framework and online trust factors: A case study on facebook," in *Digital Information Management (ICDIM), 2010 Fifth International Conference on*, July 2010, pp. 266–273.
[25] C. Wei, R. Khoury, and S. Fong, "Web 2.0 recommendation service by multi-collaborative filtering trust network algorithm," *Information Systems Frontiers*, vol. 15, no. 4, pp. 533–551, Sep. 2013.
[26] M. Li, N. Cao, S. Yu, and W. Lou, "Findu: Privacy-preserving personal profile matching in mobile social networks," *INFOCOM 2011. The 30th Conference on Computer Communications. IEEE*, pp. 2435–2443, Apr. 2011.
[27] W. Dong, V. Dave, L. Qiu, and Y. Zhang, "Secure friend discovery in mobile social networks," *INFOCOM 2011. The 30th Conference on Computer Communications. IEEE*, pp. 1647–1655, Apr. 2011.
[28] L. Guo, X. Liu, Y. Fang, and X. Li, "User-centric private matching for ehealth networks - a social perspective," in *Global Communications Conference (GLOBECOM), 2012 IEEE*, 2012, pp. 732–737.
[29] L. Guo, C. Zhang, J. Sun, and Y. Fang, "PAAS: Privacy-preserving attribute-based authentication system for ehealth networks," in *The 32nd*

IEEE International Conference on Distributed Computing Systems, ser. ICDCS 2012. Macau, China: IEEE, 2012.
[30] ——, "A privacy-preserving attribute-based authentication system for mobile health networks," *Mobile Computing, IEEE Transactions on*, vol. 13, no. 9, pp. 1927–1941, Sept 2014.
[31] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, vol. 27, no. 1, pp. 415–444, 2001.
[32] L. Guo, C. Zhang, H. Yue, and Y. Fang, "A privacy-preserving social-assisted mobile content dissemination scheme in dtns," in *The 32nd IEEE International Conference on Computer Communications*, ser. INFOCOM 2013. Turin, Italy: IEEE, 2013, pp. 2349–2357.
[33] ——, "Psad: A privacy-preserving social-assisted content dissemination scheme in dtns," *Mobile Computing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.
[34] O. Richters and T. P. Peixoto, "Trust transitivity in social networks," *CoRR*, vol. http://arxiv.org/abs/1012.1358, 2010.
[35] A. Groce and J. Katz, "A new framework for efficient password-based authenticated key exchange," in *Proceedings of the 17th ACM conference on Computer and communications security*, ser. CCS '10. New York, NY, USA: ACM, 2010, pp. 516–525.
[36] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," *Advances in Cryptology —CRYPTO 2001*, pp. 213–229, 2001.
[37] E.-J. Goh, "Encryption Schemes from Bilinear Maps," Ph.D. dissertation, Department of Computer Science, Stanford University, Sep 2007.
[38] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Mask: anonymous on-demand routing in mobile ad hoc networks," *Wireless Communications, IEEE Transactions on*, vol. 5, no. 9, pp. 2376 –2385, september 2006.
[39] D. Balfanz, G. Durfee, N. Shankar, D. Smetters, J. Staddon, and H.-C. Wong, "Secret handshakes from pairing-based key agreements," *Security and Privacy, 2003. Proceedings. 2003 Symposium on*, pp. 180 – 196, may 2003.
[40] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '91. London, UK, UK: Springer-Verlag, 1992, pp. 129–140.
[41] W. Wong, D. Cheung, B. Kao, and N. Mamoulis, "Secure knn computation on encrypted databases," *Proceedings of the 35th SIGMOD international conference on Management of data*, pp. 139–152, 2009.
[42] N.Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *INFOCOM, 2011 Proceedings IEEE*, pp. 829 –837, april 2011.
[43] W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li, "Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," in *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, ser. ASIA CCS '13. New York, NY, USA: ACM, 2013, pp. 71–82.
[44] E. Shi, T.-H. H. Chan, E. G. Rieffel, R. Chow, and D. Song, "Privacy-preserving aggregation of time-series data," *NDSS 2011*, 2011.
[45] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 2001.
[46] Facebook. [Online]. Available: http://www.sleepscholar.com/social-media-case-study/
[47] A. L. Traud, P. J. Mucha, and M. A. Porter, "Social structure of facebook networks," 2011, arXiv:1102.2166.
[48] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD trace cambridge/haggle/imote/infocom2006 (v. 2009-05-29)," May 2009.
[49] S. Trifunovic, F. Legendre, and C. Anastasiades, "Social trust in opportunistic networks," in *INFOCOM IEEE Conference on Computer Communications Workshops , 2010*, march 2010, pp. 1 –6.
[50] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Duke Univeristy, Tech. Rep., 2000.
[51] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, Jul. 2003.
[52] F. Hess, "Efficient identity based signature schemes based on pairings," *Selected Areas in Cryptography*, pp. 310–324, 2003.

Linke Guo received his B.E. degree in electronic information science and technology from Beijing University of Posts and Telecommunications in 2008. He received M.S. and Ph.D. degree in Electrical and Computer Engineering from the University of Florida in 2011 and 2014, respectively. He has been an assistant professor in the Department of Electrical and Computer Engineering, Binghamton University, State University of New York from August 2014. His research interests include network security and privacy, social networks, and applied cryptography. He has served as the Technical Program Committee (TPC) members for several conferences including IEEE INFOCOM, ICC, GLOBECOM, and WCNC. He is a member of the IEEE and ACM.

Chi Zhang received the B.E. and M.E. degrees in Electrical and Information Engineering from Huazhong University of Science and Technology, China, in 1999 and 2002, respectively, and the Ph.D. degree in Electrical and Computer Engineering from the University of Florida in 2011. He joined the University of Science and Technology of China in September 2011 as an Associate Professor of the School of Information Science and Technology. His research interests are in the areas of network protocol design and performance analysis, and network security particularly for wireless networks and social networks. He has published over 60 papers in journals such as IEEE/ACM Transactions on Networking, IEEE Journal on Selected Areas in Communications, and IEEE Transactions on Mobile Computing and in networking conferences such as IEEE INFOCOM, ICNP, and ICDCS. He has served as the Technical Program Committee (TPC) members for several conferences including IEEE INFOCOM, ICC, GLOBECOM, WCNC and PIMRC. He is the recipient of the 7th IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award.



Yuguang Fang (F'08) received a Ph.D. degree in Systems Engineering from Case Western Reserve University in January 1994 and a Ph.D degree in Electrical Engineering from Boston University in May 1997. He was an assistant professor in the Department of Electrical and Computer Engineering at New Jersey Institute of Technology from July 1998 to May 2000. He then joined the Department of Electrical and Computer Engineering at University of Florida in May 2000 as an assistant professor, got an early promotion to an associate professor with tenure in August 2003 and a professor in August 2005. He has published over 350 papers in refereed professional journals and conferences. He received the National Science Foundation Faculty Early Career Award in 2001 and the Office of Naval Research Young Investigator Award in 2002. He won the Best Paper Award at IEEE ICNP'2006. He has served on many editorial boards of technical journals including IEEE Transactions on Communications, IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing, Wireless Networks, and IEEE Wireless Communications (including the Editor-in-Chief). He is also serving as the Technical Program Co-Chair for IEEE INFOCOM'2014. He is a fellow of the IEEE.