MobiShare+: Security Improved System for Location Sharing in Mobile Online Social Networks

Jingwei Li[1], Jin Li[2], Xiaofeng Chen[3], Zheli Liu[1], and Chunfu Jia[1]*
[1] Nankai University, Tianjin, China
lijw1987@gmail.com, liuzheli1978@163.com, cfjia@nankai.edu.cn
[2] Guangzhou University, Guangzhou, China
jinli71@gmail.com
[3] Xidian University, Xi'an, China
xfchen@xidian.edu.cn

Abstract

With the development of real time localization and mobile computing technology, it becomes increasingly popular for sharing individual locations in mobile online social network (mOSN), which inevitably raises significant concerns on the privacy of users' locations. Recently, Wei et al. provided the MobiShare mechanism to enable flexible location sharing between both trusted social relations and untrusted strangers. Motivated by Wei et al.'s pioneer work, we make a further treatment on privacy-preserving location sharing in mOSN in this paper. We observe that users' real fake identities will be potentially leaked to location service provider in Wei et al.'s work, which may lead to a variety of serious attacks. In order to fix this security issue, we propose a security improved mechanism namely MobiShare+, which employs dummy queries and private set intersection protocol to prevent the OSN service provider and location service provider from learning individual information from each other. Finally, security analysis is provided to clarify that MobiShare+ is secure in terms of location privacy and social network privacy.

Keywords: Mobile cloud computing, online social network, location privacy

1 Introduction

With the development of mobile computing, the traditional social networks have gradually become fresh paradigms called mobile online social networks (mOSNs). Much like the traditional Web-based social network, mOSN also occurs in virtual community for spreading contents, increasing accessibility, and connecting users from wherever they are. Additionally, due to the "mobility" of mobile devices, mOSN has a great feature on convenient communication, which allows users to connect with their friends in social networks in real time, not bounded by time or place [3]. With well over a billion people as members, today's mOSNs pervade all aspects of our daily lives. mOSNs have grown beyond platforms for social communication and news dissemination, to indispensable tools for professional networking, social recommendations, and online content curation. Their usage has influenced today's societal and cultural issues, and changed the way we see ourselves and communicate with each other.

Location based services (LBS) is an information and entertainment service, accessible with mobile devices through the mobile network and utilizing the ability to make use of the geographical position of the mobile device [1]. LBS is a target marketer's dream come true, although some consumers see it as a scenario in which "big brother" is tracking our social, economic and personal habits. The benefits of LBS can be listed a few as follows. 1) Real estate marketing, alert homebuyers about properties in

*Corresponding author: Weijin Road 94, Tianjin, P.R. China, 300071, Tel: +86-23508656 the immediate vicinity that meet their specifications. 2) Discounts and coupons, send alerts for retail discounts within a geographic radius of an opted in consumer's location. 3) Keeping track of children and teens, parents can be alerted when kids leave or arrive at specified location, such as school or home. The benefits of LBS also makes mOSN more popular: instead of explicitly inputting their locations, recent smartphone supports various localization technologies making it easier access and share locations with their friends/strangers over the social network [8].

In tandem with the popularity of LBS and mOSN, the privacy concerns on users' locations have been raised. Specifically, since users' locations may reveal sensitive individual information such as interests, habits, health condition and so on, it desires to protect the location information against unauthorized access. This threat becomes even more serious when it comes to mOSN, in which users' physical locations are being correlated with their profiles [8]. Without a guarantee of privacy, users may be hesitant to share locations through mOSNs [2].

Recently, aiming at "flexibly" protecting users' locations, for the first time, Wei et al. [8] presented the MobiShare system which provides flexible privacy preserving location sharing in mOSN. Their work is based on dummy technique for preventing the OSN service provider and the location service provider from accessing the complete knowledge of users' identities and locations. Inspired by Wei et al.'s pioneering work [8], we attempt to make a security improvement on Wei et al.'s MobiShare mechanism [8], and provide contributions as follows.

- We observe that the real fake identity of the querying user is known by location service provider in the friends' locations query of Wei et al.'s scheme [8], which will potentially help the location service provider identify which record is true in the location database and make location dummies useless. In addition, with the real fake identity, the location service provider can obtain the friends relations and locations even if some of the them are dummies. More seriously, if we consider multiple queries without locations updates, the location service provider is able to finally obtain the topological structure of the social network and launch multiple attacks.
- Aiming at fixing this security issue, we propose a security improved system namely MobiShare+. Actually, MobiShare+ generally works in a similar way as MobiShare [8]. But unlike MobiShare [8], we empoly dummy queries in MobiShare+ besides dummy locations and identities. This allows us to protect query user's real fake identity. Moreover, we apply a private set intersection protocol between the location service provider and OSN service provider, which makes sure that anything individual will not be leaked to each other. Finally, we provide security analysis and clarify that MobiShare+ is secure in terms of location privacy and social network privacy.

This paper is organized as follows. In Section 2, we provide the system model and threat model for our mechanism. In Section 3, we briefly review Wei et al's MobiShare scheme [8] and point out its underlying security issue. The security improved MobiShare+ mechanism and its security analysis are respectively presented in Section 4 and Section 5. Finally we draw conclusion in Section 6.

2 Problem Statement

2.1 System Model

As shown in Fig. 1, we consider the scenario of location sharing in mobile online social networks, which consists of four entities.

- *Mobile Users $\mathscr{U}$* is an entity, which is able to share their own locations and flexibly query nearby friends' and strangers' locations.

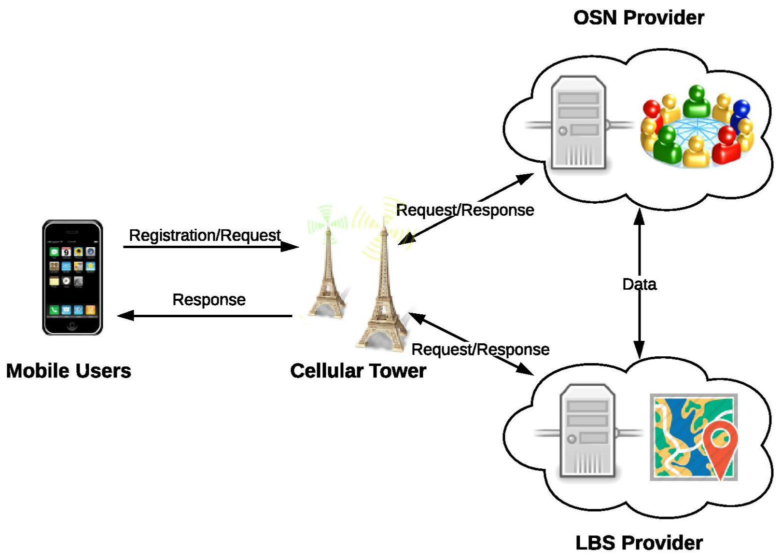


Figure 1: Architecture for Location Sharing in Mobile Online Social Network

- *Online Social Network Provider* $\mathscr{P}_{\text{OSN}}$ is an entity, which manages $\mathscr{U}$'s identity-related information (e.g., $\mathscr{U}$'s profiles and friend lists) and provides online social network service.
- *Location Service Provider* $\mathscr{P}_{\text{L}}$ is an entity, which stores $\mathscr{U}$'s location information and provides location-based services according to $\mathscr{U}$'s requests with nearby persons' locations.
- *Cellular Tower* $\mathscr{T}$ is an entity, which helps $\mathscr{U}$ communicate with $\mathscr{P}_{\text{OSN}}$ and $\mathscr{P}_{\text{L}}$.

The challenges in location sharing in mobile online social network are manifold: First, how to update $\mathscr{U}$'s location in a privacy-preserving way when arrive at a new place? Second, how to conduct privacy-preserving location query to obtain nearby friends' and strangers' current locations? We formalize these problems as follows.

Suppose $\mathscr{I} = \{\text{ID}_1, \text{ID}_2, \dots, \text{ID}_n\}$ is the identity set of all the users involved in the location sharing service, and a social network graph $\mathscr{G} = (V, E)$ on $\mathscr{I}$ has been stored at $\mathscr{P}_{\text{OSN}}$, where $V \subseteq \mathscr{I}$ is a set of identity vertices and $E \subseteq V \times V$ is a set of edges in $\mathscr{G}$. Without privacy concerns, we assume a location database in the form of $\{(\text{ID}, (x,y), df_{\text{ID}}, ds_{\text{ID}})\}_{\text{ID} \in \mathscr{I}}$ is maintained by $\mathscr{P}_{\text{L}}$, where ID is user's identity, (x,y) is his/her current location, df_{ID} is the threshold distance within which such a user is willing to share location with his/her friends, and ds_{ID} is the threshold distance within which he/she agrees to share location with strangers.

Let $(\text{ID}_i, (x_i, y_i), df_i, ds_i)$ be any valid record in location database, $\mathsf{dist}(\cdot,\cdot)$ be distance function and $\mathsf{min}(\cdot,\cdot)$ return the minimum value in its inputs. We firstly define the general goal as follows and then extend it to the privacy-preserving manner along with the associated privacy requirements in Section 2.3.

The first challenge can be modeled as how to generate/update an appropriate entry in the form of $(\text{ID}, (x,y), df_{\text{ID}}, ds_{\text{ID}})$ in location database so as to satisfy desired security goals.

The second issue on location query can be formalized in two sides.

- For friends' locations query, a user with identity ID can submit a query in the form of (ID, qf) to obtain all his/her nearby friends' locations $\{(\text{ID}_i, (x_i, y_i))\}$ satisfying the restrictions $(\text{ID}, \text{ID}_i) \in \mathscr{G}.E$ and $\mathsf{dist}((x,y),(x_i,y_i)) \le \mathsf{min}(qf, df_i)$, where qf is the distance threshold in friends' locations query and (x,y) is user ID's current location.

- For strangers' locations query, a user with identity ID can submit a query in the form of (ID, qs) to obtain all the nearby strangers' locations $\{(\mathrm{ID}_i, (x_i, y_i))\}$ with $\mathsf{dist}((x,y),(x_i,y_i)) \leq \mathsf{min}(qs, ds_i)$, where qs is the distance threshold in strangers' locations query and (x,y) is user ID's current location.

2.2 System Overview

Based on the proposed system model, we give an overview of the location sharing mechanism which involves five phases as follows.

- *Location-based Service Registration.* Initially, users register for the LBS at $\mathscr{P}_{\mathrm{OSN}}$. Specifically, they have to specify their profiles and individual preferences (e.g., access policy for his/her shared location) which will be kept as a record at OSN provider.
- *Cellular Registration.* Before using the LBS, users have to register their profiles at local cellular tower.
- *Location Updates.* When arrives at a new place, user updates his/her location information at LBS provider.
- *Friends' Locations Query.* When a user wants to know his/her nearby friends' current locations, he/she queries the OSN provider and LBS provider through local cellular tower, and receives the location information of friends whose specified access control setting is satisfied by the querying user.
- *Strangers' Locations Query.* When a user wants to know nearby strangers' current locations, he/she queries the OSN provider and LBS provider through local cellular tower, and receives the location information of someone whose specified access control setting is satisfied by the querying user.

2.3 Adversary Model

Within the four entities involved, the cellular tower is assumed as a trusted entity. Here we are concerned with security risks with respect to OSN and LBS providers, and mobile users.

- *OSN and LBS Providers.* We consider "honest-but-curious" service providers. That is, both OSN and LBS providers are supposed to follow our proposed protocol in general but try to find out as much "private" information as possible. Moreover, as with the assumption in [8], just either of the OSN and LBS providers is assumed to be compromised and controlled by adversary seeking to link users' identities to their locations. This assumption is reasonable because it is unlikely that two service providers operated by independent organizations can be controlled by the same adversary.
- *Dishonest Mobile Users.* Dishonest mobile users would try to access the location information outside the scope of their access privileges. To do so, unauthorized users may attempt to collude with either of the OSN and LBS providers to change their individual preferences.

As with MobiShare [8] the first security requirement considered in this paper is *location privacy* which prevents OSN or LBS provider from accessing users' real locations, even if either of them colludes with dishonest and unauthorized users. Moreover, unlike the previous work which will potentially leak partial information of the social network to LBS provider, we consider another security requirement namely *Social Network Privacy* which prevents LBS provider from precisely accessing the social network, even if it colludes with dishonest users.

Symbol	Description
ID	User's social network identifier, used as his/her identity
$(x_{\mathrm{ID}}, y_{\mathrm{ID}})$	User ID's real location
FID	Fake identifier including real and dummy ones
df_{ID}	User ID's friend-case threshold distance
ds_{ID}	User ID's stranger-case threshold distance
qf	Distance threshold in friends' locations query
qs	Distance threshold in strangers' locations query
$k_{\mathscr{U}}$	A symmetric key shared by user and his/her friends
$\mathscr{G}$	A social network graph stored at $\mathscr{P}_{\mathrm{OSN}}$
$pk_{\mathscr{U}}$	User's public key
$sk_{\mathscr{U}}$	User's private key
$pk_{\mathscr{P}_{\mathrm{L}}}$	Location service provider's public key
$sk_{\mathscr{P}_{\mathrm{L}}}$	Location service provider's private key
$pk_{\mathscr{C}\mathscr{T}}$	Cellular tower's public key
$sk_{\mathscr{C}\mathscr{T}}$	Cellular tower's private key

Table 1: Summary of Notations

3 MobiShare Revisit

3.1 Notations

We summarize the notations used in this paper in Table 1.

3.2 A Brief Description of MobiShare

Before introducing our detailed mechanism, we first revisit the MobiShare mechanism in [8], which shares the same aim with ours, and explore underlying security risks in such a scheme.

In MobiShare, users have to first register at both $\mathscr{P}_{\mathrm{OSN}}$ and $\mathscr{C}\mathscr{T}$ for enjoying the location-sharing service. For simplicity, we omit the both stages in our description and just assume user preference record $(\mathrm{ID}, df_{\mathrm{ID}}, ds_{\mathrm{ID}})$ has been stored at $\mathscr{C}\mathscr{T}$ and $\mathscr{P}_{\mathrm{OSN}}$. The other three stages are briefly elaborated as follows.

- *Location Update.* In order to upload or update location, a user $\mathscr{U}$ with identity ID will get his/her current location through GPS or cellular geolocation, and send this real update to $\mathscr{C}\mathscr{T}$. $\mathscr{C}\mathscr{T}$ splits the real update into two parts (i.e., the identity part and the location part), and imposes dummy updates on both of them. Specifically, 1) To hide real identity, $\mathscr{C}\mathscr{T}$ generates k fake IDs (denoted by FID), one of which (denoted by $\mathrm{FID}_{\mathrm{Real}}$ without loss of generality) is used to replace ID in real location update, and then sends the dummy identity part in the form of $(\mathrm{ID}, \mathrm{FID}_{\mathrm{Real}}, \{\mathrm{FID}_i\}_{i=1}^{k-1})$ to $\mathscr{P}_{\mathrm{OSN}}$. 2) To hide real location, $\mathscr{C}\mathscr{T}$ generates $k-1$ fake locations, denoted by $\{(x_i, y_i)\}_{i=1}^{k-1}$, and sends $k-1$ dummy location $(\{\mathrm{FID}_i, (x_i, y_i), str_i, df_i, ds_i\}_{i=1}^{k-1})$ and the real location update $(\mathrm{FID}_{\mathrm{Real}}, (x,y), \mathsf{Enc}_{k_{\mathscr{U}}}(x,y), df_{\mathrm{ID}}, ds_{\mathrm{ID}})$ to $\mathscr{P}_{\mathrm{L}}$, where (x,y) is user's real location, $k_{\mathscr{U}}$ is a symmetric key shared among all the user's friends, df_{ID} is friend-case threshold distance, ds_{ID} is stranger-case threshold distance, and $(x_i, y_i), str_i, df_i, ds_i$ are random values imitating $(x,y), \mathsf{Enc}_{k_{\mathscr{U}}}(x,y), df_{\mathrm{ID}}, ds_{\mathrm{ID}}$ respectively.
- *Friends' Locations Query.* To query the locations of friends within a certain range, user submits query in the form of (ID, qf) to the nearest $\mathscr{C}\mathscr{T}$, which attaches the encrypted tower identity and

sequence number to prevent replay attack, and forwards the appended query to $\mathscr{P}_{\text{OSN}}$. $\mathscr{P}_{\text{OSN}}$ substitutes the "real fake identity" FID_{Real} for user's real identity ID, and retrieve a friend list $\mathsf{FIDlist}$ consisting of the fake identities of all the user's friends. That is $\mathsf{FIDlist}$ includes the fake identities used in each friend's latest real and dummy location updates. The both components (i.e., FID and $\mathsf{FIDlist}$) as well as the encrypted tower identity and sequence number are sent to $\mathscr{P}_{\text{L}}$. On the reception of the query, $\mathscr{P}_{\text{L}}$ checks which fake identities in $\mathsf{FIDlist}$ are within df_{ID} threshold away from FID_{Real}. For each of the nearby identity FID, $\mathscr{P}_{\text{L}}$ enforces access control based on FID's friend-case threshold distance. After finishing the distance computation and access control enforcement, $\mathscr{P}_{\text{L}}$ replies the encryption of retrieved fake identities, their corresponding encrypted locations and the sequence number to $\mathscr{P}_{\text{OSN}}$, which just appends a mapping entry for each of the user's friends to the message and forwards it to $\mathscr{C}\mathscr{T}$. $\mathscr{C}\mathscr{T}$ finally decrypts and sends the original identities and their corresponding encrypted locations to user. Note that the symmetric key is shared among all the friends which makes sure that user can decrypt the encrypted location and obtain the plain one.

- *Strangers' Locations Query.* The location query for strangers is similar to that in the friend case with only two differences. One is that the locations of queried identities are transfered in plain and not required to be in encrypted form. This is because unlike friends' locations query in which all the friends share a symmetric key, user has nothing shared with strangers and he/she cannot perform decryption. The second difference is $\mathscr{P}_{\text{OSN}}$ does not need to retrieve the friend list $\mathsf{FIDlist}$ during query submission but it just appends the mapping entry with $\mathscr{P}_{\text{L}}$'s reply as what it does in friends' locations query. The detailed procedure can be referred to [8].

3.3 Observations

Generally, Mobishare [8] utilizes the dummy technique to achieve a valuable goal, that is providing a flexible way for privacy-preserving location sharing for both trusted friends and untrusted strangers, simultaneously supporting range query and user-defined access control. However, from technical prospective, we specify that in the original Mobishare mechanism [8], the dummy entries used at $\mathscr{P}_{\text{L}}$ can be easily distinguished, which may result in underlying security risk.

Recall that in Mobishare [8], for each real location entry $(\text{FID}_{\text{Real}}, (x,y), \mathsf{Enc}_{k_{\mathscr{U}}}(x,y), df_{\text{ID}}, ds_{\text{ID}})$, $k-1$ dummy entries $(\text{FID}_i, (x_i,y_i), str_i, df_i, ds_i)$ are maintained at $\mathscr{P}_{\text{L}}$ to prevent $\mathscr{P}_{\text{L}}$ from knowing whether the corresponding relation $(\text{FID}, (x,y))$ is real. This technique works in location update and $\mathscr{P}_{\text{L}}$ has nearly the probability of $\frac{1}{k}$ to choose a real corresponding relation. Nevertheless, considering a user's location query, the dummy entries can be easily distinguished with $\mathscr{P}_{\text{OSN}}$'s forwarded query. Specifically, whatever in friends' or strangers' locations query, $\mathscr{P}_{\text{OSN}}$ translates user's identity ID to the fake one FID_{Real} and sends it to $\mathscr{P}_{\text{L}}$. This will help $\mathscr{P}_{\text{L}}$ identify which location entry is real in its local database.

More seriously, $\mathscr{P}_{\text{L}}$ can record the user's identity in query for each time, each of which indicates an underlying real location entry at $\mathscr{P}_{\text{L}}$. If we consider queries for multiple times but without any location updates[1], $\mathscr{P}_{\text{L}}$ will finally obtain the topological structure for the whole social network. With such topology, many kinds of attacks can be launched. For example, $\mathscr{P}_{\text{L}}$ is able to find the most sociable person and identify his/her real location. $\mathscr{P}_{\text{L}}$ can also use the recent social network de-anonymization techniques [9][5][7] to obtain original social network.

[1] It is reasonable because we can consider many users simultaneously query friends' or strangers' locations, and no updates are required in such a time slot

4 MobiShare+ Mechanism

4.1 Cryptographic Tool: Paillier Cryptosystem

Before providing the MobiShare+ mechanism in detail, we firstly introduce the Paillier cryptosystem [6] upon which the proposed mechanism is based.

The Paillier cryptosystem is a probabilistic asymmetric algorithm for public key cryptography. It is based on the decisional composite residuosity assumption which is an intractability hybpthesis in the problem of computing n-th residue classes. We provide the Paillier cryptosystem as follows.

- $\mathsf{Gen}_{\mathsf{Pail}}$: An entity chooses two primes p and q, and computes $N = pq$ and $\lambda = \mathsf{lcm}(p-1, q-1)$, where $\mathsf{lcm}(\cdot,\cdot)$ computes the lowest common multiple of its inputs. It then selects a random $g \in \mathbb{Z}^*_{N^2}$ such that $\mathsf{gcd}((\mathsf{L}(g^\lambda \mod N^2), N)) = 1$, where $\mathsf{L}(x) = \frac{(x-1)}{N}$ and $\mathsf{gcd}(\cdot,\cdot)$ computes the greatest common divisor of its inputs. Then the public and private keys of Paillier cryptosystem are (N, g) and λ respectively.

- $\mathsf{Enc}_{\mathsf{Pail}}$: Let $m \in \mathbb{Z}_N$ be a plaintext to be encrypted and $r \in \mathbb{Z}_N$ be a random number. The ciphertext of Paillier cryptosystem is given by $\mathsf{Enc}_{\mathsf{Pail}}(m \mod N, r \mod N) = g^m r^N \mod N^2$.

- $\mathsf{Dec}_{\mathsf{Pail}}$: Given a ciphertext $c \in \mathbb{Z}_{N^2}$, the corresponding plaintext can be derived as $\mathsf{Dec}_{\mathsf{Pail}}(c) = \frac{\mathsf{L}(c^\lambda \mod N^2)}{\mathsf{L}(g^\lambda \mod N^2)} \mod N$.

The most desirable feature of Paillier cryptosystem is its homomorphism. More precisely, for any $m_1, m_2, r_1, r_2 \in \mathbb{Z}_N$, we have equations: $\mathsf{Enc}_{\mathsf{Pail}}(m_1, r_1)\mathsf{Enc}_{\mathsf{Pail}}(m_2, r_2) = \mathsf{Enc}_{\mathsf{Pail}}(m_1 + m_2, r_1 r_2) \mod N^2$ and $(\mathsf{Enc}_{\mathsf{Pail}}(m_1, r_1))^{m_2} = \mathsf{Enc}_{\mathsf{Pail}}(m_1 m_2, r_1^{m_2}) \mod N^2$.

4.2 Mechanism Description

We provide the MobiShare+ mechanism in detail as follows. Note that all the fake identities in MobiShare+ are assumed to be picked from an integer field.

4.2.1 Location-based Service Registration

Initially, user registers for the service at $\mathscr{P}_{\mathrm{OSN}}$. During the registration, a social network identifier ID is assigned to the user and his/her friend relations at $\mathscr{G}$ is updated. Meanwhile, user shares his/her public key $pk_{\mathscr{U}}$ with $\mathscr{P}_{\mathrm{OSN}}$ and defines an access control setting consisting of df_{ID} and ds_{ID} to restrict location sharing. Finally, user keeps a record of his/her social network identifier ID, while $\mathscr{P}_{\mathrm{OSN}}$ stores an entry $(\mathrm{ID}, pk_{\mathscr{U}}, df_{\mathrm{ID}}, ds_{\mathrm{ID}})$ at its local subscriber database.

4.2.2 Cellular Registration

Similar to MobiShare, when a user handset connects to a cellular tower $\mathscr{CT}$, three steps are performed as follows.

Step 1 User sends an authentication request in the form of $(\mathrm{ID}, ts, \mathsf{Sig}_{sk_{\mathscr{U}}}(\mathrm{ID}, ts))$ to $\mathscr{CT}$ which directly forwards the message to $\mathscr{P}_{\mathrm{OSN}}$, where ts is current timestamp and $\mathsf{Sig}_{sk_{\mathscr{U}}}(\cdot)$ is a signature with user's private key $sk_{\mathscr{U}}$.

Step 2 $\mathscr{P}_{\mathrm{OSN}}$ finds the appropriate entry $(\mathrm{ID}, pk_{\mathscr{U}}, df_{\mathrm{ID}}, ds_{\mathrm{ID}})$ in subscriber database and verifies the correctness of $\mathsf{Sig}_{sk_{\mathscr{U}}}(\mathrm{ID}, ts))$ with $pk_{\mathscr{U}}$. If it is passed, a reply $(\mathrm{ID}, df_{\mathrm{ID}}, ds_{\mathrm{ID}})$ is sent to and stored at $\mathscr{CT}$. Note that if ID is the first user registered at this $\mathscr{CT}$, a network address of $\mathscr{P}_{\mathrm{L}}$ is included in this reply as well.

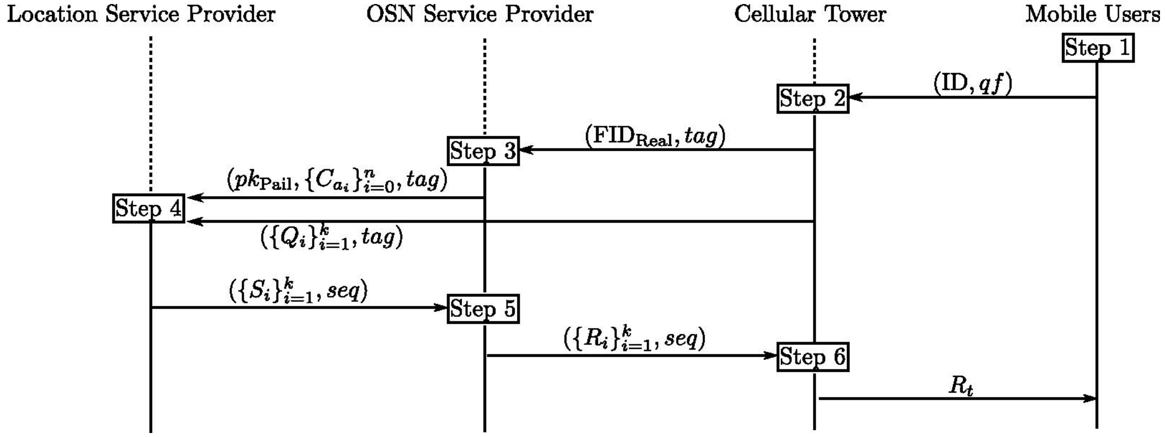

Location Service Provider
OSN Service Provider
Cellular Tower
Mobile Users
Step 1
(ID, qf)
Step 2
($\mathrm{FID}_{\mathrm{Real}}$, tag)
Step 3
(pk_{Pail}, $\{C_{a_i}\}_{i=0}^{n}$, tag)
Step 4
($\{Q_i\}_{i=1}^{k}$, tag)
($\{S_i\}_{i=1}^{k}$, seq)
Step 5
($\{R_i\}_{i=1}^{k}$, seq)
Step 6
R_t

Figure 2: Querying Friends' Locations

Step 3 $\mathscr{C}\mathscr{T}$ finally returns an OK signal to user to indicate authentication success.

4.2.3 Location Updates

As with MobiShare, we employ the dummy techniques at $\mathscr{P}_{\mathrm{L}}$ and stores the mapping entries at $\mathscr{P}_{\mathrm{OSN}}$. The detailed steps are as follows.

Step 1 To update his/her location, a user $\mathscr{U}$ with identity ID uploads $(\mathrm{ID},(x,y),\mathsf{SKEnc}_{k_{\mathscr{U}}}(x,y))$ to $\mathscr{C}\mathscr{T}$.

Step 2 $\mathscr{C}\mathscr{T}$ randomly generates k fake identities FIDs and let one of which be real fake identity $\mathrm{FID}_{\mathrm{Real}}$. $\mathscr{C}\mathscr{T}$ sends the mapping entry $(\mathrm{FID}_{\mathrm{Real}},\mathrm{ID},\{\mathrm{FID}_i\}_{i=1}^{k-1})$ to $\mathscr{P}_{\mathrm{OSN}}$. Note that the appropriate entry of local user information table at $\mathscr{C}\mathscr{T}$ is updated as $(\mathrm{ID},\mathrm{FID}_{\mathrm{Real}},df_{\mathrm{ID}},ds_{\mathrm{ID}})$.

Step 3 Meanwhile, $\mathscr{C}\mathscr{T}$ uses the fake identities to generate dummy location updates. Specifically, for each FID_i, $\mathscr{C}\mathscr{T}$ picks a random string str_i, a random location (x_i,y_i) and two random threshold values df_i and ds_i, and sends $(\mathrm{FID}_i,str_i,df_i,ds_i)$ to $\mathscr{P}_{\mathrm{L}}$; for the real fake identity $\mathrm{FID}_{\mathrm{Real}}$, $\mathscr{C}\mathscr{T}$ reads the appropriate entry $(\mathrm{ID},df_{\mathrm{ID}},ds_{\mathrm{ID}})$ in its local user information table, and sends the message $(\mathrm{FID}_{\mathrm{Real}},(x,y),\mathsf{SKEnc}_{k_{\mathscr{U}}}(x,y),df_{\mathrm{ID}},ds_{\mathrm{ID}})$ to $\mathscr{P}_{\mathrm{ID}}$.

4.2.4 Friends' Locations Query

Unlike the MobiShare which directly transfers selections in plain, we utilize a set intersection protocol between $\mathscr{P}_{\mathrm{OSN}}$ and $\mathscr{P}_{\mathrm{L}}$ to preserve privacy for each other (Fig. 2 shows the message transmission in this stage).

Step 1 User with identity ID submits a friends' locations query (ID,qf) to $\mathscr{C}\mathscr{T}$.

Step 2 $\mathscr{C}\mathscr{T}$ reads the appropriate entry $(\mathrm{ID},\mathrm{FID}_{\mathrm{Real}},df_{\mathrm{ID}},ds_{\mathrm{ID}})$ in the local user information table, and generate k dummy queries $\{(\mathrm{FID}_i,qf_i)\}_{i=1}^{k-1}$ and $(\mathrm{FID}_{\mathrm{Real}},qf)$, where FID_i is picked randomly from the already used random subscribers' fake identities and qf_i is random value. Finally, after embeding a query identifier $tag = \mathsf{PKEnc}_{pk_{\mathscr{P}_{\mathrm{L}}}}(\mathrm{ID}_{\mathscr{C}\mathscr{T}},seq)$ into user's query, $\mathscr{C}\mathscr{T}$ sends the k dummy queries $\{Q_i\}_{i=1}^{k}$ in random order as well as the tag. Note that without loss of generality, we assume that the real dummy query $(\mathrm{FID}_{\mathrm{Real}},qf)$ is at t-th place (i.e., $Q_t = (\mathrm{FID}_{\mathrm{Real}},qf)$). Meanwhile $\mathscr{C}\mathscr{T}$ also sends $(\mathrm{FID}_{\mathrm{Real}},tag)$ to $\mathscr{P}_{\mathrm{OSN}}$.

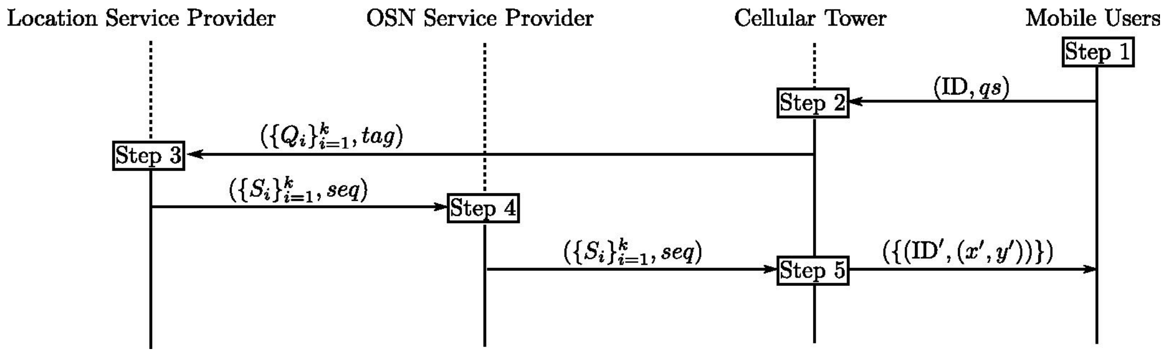

Location Service Provider
OSN Service Provider
Cellular Tower
Mobile Users
Step 1
(ID, qs)
Step 2
$(\{Q_i\}_{i=1}^{k}, tag)$
Step 3
$(\{S_i\}_{i=1}^{k}, seq)$
Step 4
$(\{S_i\}_{i=1}^{k}, seq)$
Step 5
$(\{(\text{ID}', (x', y'))\})$

Figure 3: Querying Strangers' Locations

Step 3 $\mathscr{P}_{\text{OSN}}$ selects a friend identity set consisting of all the friend identities ID$'$ with $(\text{ID},\text{ID}') \in \mathscr{G}.E$, and transforms it to real fake identity set $\mathscr{F}$ according to identity mapping entries. $\mathscr{P}_{\text{OSN}}$ builds an n-degree polynomial as $f_{\mathscr{F}}(x) = \prod_{\text{FID}_{\text{Real}} \in \mathscr{F}} (x - \text{FID}_{\text{Real}}) \prod_{i=1}^{n-|\mathscr{F}|} (x - num_i) = \sum_{i=0}^{n} a_i x^i$, where n is a predefined integer larger than the degree of the social network graph and num_i is random identity. Finally, after running $\mathsf{Gen}_{\text{Pail}}$ and obtaining pk_{Pail} and sk_{Pail}, $\mathscr{P}_{\text{OSN}}$ sends $(pk_{\text{Pail}}, \{C_{a_i}\}_{i=0}^{n}, tag)$ to $\mathscr{P}_{\text{L}}$, where $C_{a_i} = \mathsf{Enc}_{\text{Pail}}(pk_{\text{Pail}}, a_i)$.

Step 4 Upon receiving both $(\{(\text{FID}_i, qf_i)\}_{i=1}^{k-1}, (\text{FID}_{\text{Real}}, qf), tag)$ and $(pk_{\text{Pail}}, \{\mathsf{Enc}_{\text{Pail}}(pk_{\text{Pail}}, a_i)\}_{i=0}^{n}, tag)$, $\mathscr{P}_{\text{L}}$ checks the validity of the query by decrypting $tag = \mathsf{PKEnc}_{pk_{\mathscr{P}_{\text{L}}}}(\text{ID}_{\mathscr{C}\mathscr{T}}, seq))$ and examining the identity of $\mathscr{C}\mathscr{T}$ and the sequence number seq. If valid, it continues to build k sets $\{S_i\}_{i=1}^{k}$ in the following manner: For each entry $(\text{FID}', (x', y'), \mathsf{SKEnc}_{k_{\mathscr{U}}}(x', y'), df', ds')$ stored in the location database, check whether $\mathsf{dist}((x_i, y_i), (x', y')) \leq \mathsf{min}(qf_i, df')$, if so compute $y' = \mathsf{Enc}_{\text{Pail}}(pk_{\text{Pail}}, r' f_{\mathscr{F}}(\text{FID}') + \text{FID}')$ with the homomorphism of Paillier cryptosystem and set $S_i = S_i \cup \{(y', \mathsf{SKEnc}_{k_{\mathscr{U}}}(x', y'))\}$, where (x_i, y_i) is the corresponding location of FID_i and r' is a random integer. Finally $\mathscr{P}_{\text{L}}$ replies $(\{S_i\}_{i=1}^{k}, seq)$ to $\mathscr{P}_{\text{OSN}}$.

Step 5 Upon receiving $(\{S_i\}_{i=1}^{k}, seq)$, $\mathscr{P}_{\text{OSN}}$ builds k sets $\{R_i\}_{i=1}^{k}$ as follows. For each item in the set $(y', \mathsf{SKEnc}_{k_{\mathscr{U}}}(x', y')) \in S_i$, $\mathscr{P}_{\text{OSN}}$ executes $x' = \mathsf{Dec}_{\text{Pail}}(sk_{\text{Pail}}, y')$. If $x' \notin \mathscr{F}$, discard x'; otherwise read its corresponding identity ID and set $R_i = R_i \cup \{(\text{ID}, \mathsf{SKEnc}_{k_{\mathscr{U}}}(x', y'))\}$. Finally reply $(\{R_i\}_{i=1}^{k}, seq)$ to $\mathscr{C}\mathscr{T}$.

Step 6 $\mathscr{C}\mathscr{T}$ checks whether the seq in reply corresponds to the sequence number it previously sent. If so it sends R_t back to user.

4.2.5 Strangers' Locations Query

The strangers' locations query performs in a simialr manner with friends' locations query, but does not require $\mathscr{P}_{\text{OSN}}$ to find friends and build encrypted polynomial (Fig. 3 shows the message transmission in this stage).

Step 1 User with identity ID submits a friends' locations query (ID, qs) to $\mathscr{C}\mathscr{T}$.

Step 2 $\mathscr{C}\mathscr{T}$ reads the appropriate entry $(\text{ID}, \text{FID}_{\text{Real}}, df_{\text{ID}}, ds_{\text{ID}})$ in the local user information table, and generates k dummy queries $\{(\text{FID}_i, qs_i)\}_{i=1}^{k-1}$ and $(\text{FID}_{\text{Real}}, qs)$, where FID_i is picked from the already used random subscribers' fake identities and qs_i is random value. Finally, after embeding a query identifier $tag = \mathsf{PKEnc}_{pk_{\mathscr{P}_{\text{L}}}}(\text{ID}_{\mathscr{C}\mathscr{T}}, seq)$ into user's query, $\mathscr{C}\mathscr{T}$ sends the k dummy queries

$\{Q_i\}_{i=1}^{k}$ in random order as well as the *tag*. Note that without loss of generality, we assume that the real dummy query $(\mathrm{FID}_{\mathrm{Real}}, qs)$ is at t-th place (i.e., $Q_t = (\mathrm{FID}_{\mathrm{Real}}, qs)$).

Step 3 Upon receiving $(\{(\mathrm{FID}_i, qs_i)\}_{i=1}^{k-1}, (\mathrm{FID}_{\mathrm{Real}}, qs), tag)$, $\mathscr{P}_{\mathrm{L}}$ checks the validity of the query by decrypting $tag = \mathsf{PKEnc}_{pk_{\mathscr{P}_{\mathrm{L}}}}(\mathrm{ID}_{\mathscr{CT}}, seq))$ and examining the identity of $\mathscr{CT}$. If valid, it builds k sets $\{S_i\}_{i=1}^{k}$ in the following manner: For each entry $(\mathrm{FID}', (x', y'), \mathsf{SKEnc}_{k_{\mathscr{U}}}(x', y'), df', ds')$ stored in the location database, check whether $\mathsf{dist}((x_i, y_i), (x', y')) \leq \mathsf{min}(qs_i, ds')$, if so set $S_i = S_i \cup \{(\mathrm{FID}', \mathsf{PKEnc}_{pk_{\mathscr{CT}}}(x', y'))\}$, where (x_i, y_i) is the corresponding location of fake identity FID_i. Finally $\mathscr{P}_{\mathrm{L}}$ sends $(\{S_i\}_{i=1}^{k}, seq)$ to $\mathscr{P}_{\mathrm{OSN}}$.

Step 4 Upon receiving $(\{S_i\}_{i=1}^{k}, seq)$, $\mathscr{P}_{\mathrm{OSN}}$ proceeds as follows. For each $(\mathrm{FID}', \mathsf{PKEnc}_{pk_{\mathscr{CT}}}(x', y')) \in S_i$, it replaces FID' by its corresponding identity ID' and remove the duplicated item (i.e., two items have the same identity) in S_i, where $i = 1, 2, \ldots, k$. Finally $\mathscr{P}_{\mathrm{OSN}}$ replies the processed $(\{S_i\}_{i=1}^{k}, seq)$ to $\mathscr{CT}$.

Step 5 $\mathscr{CT}$ checks whether the *seq* in reply corresponds to the sequence number it previously sent. If so for each item $(\mathrm{ID}', \mathsf{PKEnc}_{pk_{\mathscr{CT}}}(x', y'))$ in S_t, it decrypts the encrypted location and returns $(\mathrm{ID}', (x', y'))$ back to user.

5 Security Analysis

Recall that $\mathscr{P}_{\mathrm{OSN}}$ and $\mathscr{P}_{\mathrm{L}}$ are both assumed to be "honest-but-curious" and cannot be controlled by adversary at the same time. We provide the security analysis according to the two security concerns mentioned in Section 2.3.

- *Location Privacy.* The location privacy is threated by $\mathscr{P}_{\mathrm{OSN}}$ or $\mathscr{P}_{\mathrm{L}}$ colluding with dishonest users. For the former, the chance of accessing users' locations is when receiving the response from $\mathscr{P}_{\mathrm{L}}$ in friends and strangers' locations query. Note that in these replies during the both stages, the real locations are protected by symmetric/asymmetric encryption scheme, which will not leak any information to $\mathscr{P}_{\mathrm{OSN}}$. For the latter, though all the users' locations are stored in plain at $\mathscr{P}_{\mathrm{L}}$, a k-anonymity technique is applied to restrict that $\mathscr{P}_{\mathrm{L}}$ has at most the probability of $\frac{1}{k}$ to guess which is user's real location.

- *Social Network Privacy.* In the MobiShare+, the privacy of social network is prevented from $\mathscr{P}_{\mathrm{L}}$ through the private two set interaction protocol [4] in friends' locations query. Specifically, we let $\mathscr{P}_{\mathrm{OSN}}$ firstly find out all the fake identity FID_i, of which the corresponding identity satisfies $(\mathrm{ID}, \mathrm{ID}_i) \in \mathscr{G}.E$. With these fake identities as zero point, we build a polynomial and encrypt the coefficients of polynomial with Paillier cryptosystem [6]. This allows us to hide the fake identities from $\mathscr{P}_{\mathrm{L}}$. Moreover, unlike MobiShare [8] which directly forwards user's real fake identity to $\mathscr{P}_{\mathrm{L}}$, we apply $k-1$ dummy queries to prevent $\mathscr{P}_{\mathrm{L}}$ from knowing which is user's real fake identity. Based on the analysis on the two points above, the relations between user's fake identity and his frineds' fake identities are hidden as well. Finally we can conclude that the privacy of social network is preserved.

6 Conclusion

In this paper, we make a security improvement on Wei et al.'s MobiShare [8], and propose the MobiShare+ mechanism. Compared with MobiShare [8], MobiShare+ employs dummy queries and private set intersection protocol between the location service and OSN service provider, and fully protects the social network against the location service provider.

References

[1] Top benefits of location-based services, April 2010. `http://www.pingmobile.com/blog/top-benefits-of-location-based-services/`.

[2] L. Barkhuus and A. K. Dey. Location-based services for mobile telephony: a study of users' privacy concerns. In *Proc. of the 2003 IFIP TC13 International Conference on Human-Computer Interaction (INTERACT'03), Zurich, Switzerland*, pages 702–712. IOS Press, September 2003.

[3] H. T. Dinh, C. Lee, D. Niyato, and P. Wang. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless Communications and Mobile Computing*. Article first published online: 11 Ocotober 2011.

[4] M. Freedman, K. Nissim, and B. Pinkas. Efficient private matching and set intersection. In *Proc. of the 2004 International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'04), Interlaken, Switzerland, LNCS*, volume 3027, pages 1–19. Springer-Verlag, May 2004.

[5] A. Narayanan and V. Shmatikov. De-anonymizing social networks. In *Proc. of the 30th IEEE Symposium on Security and Privacy (IEEE S&P'09), Oakland, California, USA*, pages 173–187. IEEE, May 2009.

[6] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Proc. of the 17th international conference on Theory and application of cryptographic techniques (EUROCRYPT'99), Prague, Czech Republic, LNCS*, volume 1592, pages 223–238. Springer-Verlag, May 1999.

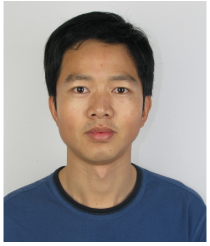
[7] G. Wang, Q. Liu, F. Li, S. Yang, and J. Wu. Outsourcing privacy-preserving social networks to a cloud. In *Proc. of the 32nd IEEE International Conference on Computer Communications (INFOCOM'13), Turin, Italy*, pages 2886–2894. IEEE, April 2013.

[8] W. Wei, F. Xu, and Q. Li. MobiShare: Flexible privacy-preserving location sharing in mobile online social networks. In *Proc. of the 31st IEEE International Conference on Computer Communications (INFOCOM'12), Orlando, Florida, USA*, pages 2616–2620. IEEE, March 2012.

[9] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In *Proc. of the 24th IEEE International Conference on Data Engineering (ICDE'08), Cancun, Mexico*, pages 506–515. IEEE, April 2008.

Author Biography

Jingwei Li received his B.S. on Mathematics in 2005 from the Hebei University of Technology, China. Now he is a PhD candidate in computer technology in Nankai University. His current interests include applied cryptography, cloud security.



Jin Li received his B.S. (2002) and M.S. (2004) from Southwest University and Sun Yat-sen University, both in Mathematics. He got his Ph.D degree in information security from Sun Yat-sen University at 2007. Currently, he works at Guangzhou University. His research interests include Applied Cryptography and Security in Cloud Computing (secure outsourcing computation and cloud storage). He served as a senior research associate at Korea Advanced Institute of Technology (Korea) and Illinois Institute of Technology (U.S.A.) from 2008 to 2010, respectively. He has published more than 40 papers in international conferences and journals, including IEEE INFOCOM, IEEE Transaction on Parallel and Distributed Computation, IEEE Transaction on Information Forensics and Security, ESORICS etc. He also served as TPC committee for many international conferences on security. He received a National Science Foundation of China (NSFC) Grant for his research on secure outsourcing computation in cloud computing. He was selected as one of science and technology new stars in Guangdong province.

Xiaofeng Chen received his B.S. and M.S. on Mathematics in Northwest University, China. He got his Ph.D degree in Cryptography in Xidian University at 2003. Currently, he works at Xidian University as a professor. His research interests include applied cryptography and cloud security.

Zheli Liu obtained his B.E. degree on System Architecture in 2005 and his Ph.D. degree on Computer Application in 2009 from the Jilin University, China. Now he is a lecturer in Nankai University, China. His current interests include cryptography, card operation system.

Chunfu Jia obtained his Ph.D. degree on Engineering in Nankai University in 1996. He has finished his post-doctor research in University of Science and Technology of China. Now he is a professor in Nankai University, China. His current interests include computer system security, network security, trusted computing, malicious code analysis, etc. Till now, he has contributed more than 90 papers, more than 30 of which have been indexed by EI or SCI.