A New Mobile Online Social Network based Location Sharing with Enhanced Privacy Protection

Junggab Son*, Donghyun Kim*, Rahman Tashakkori†, Alade O. Tokuta‡ and Heekuck Oh§
*Department of Computer Science, Kennesaw State University, Marietta, GA 30060, USA.
Email: {json, donghyun.kim}@kennesaw.edu
†Department of Computer Science, Appalachian State University, Boone, NC 28608, USA.
Email: tashakkorir@appstate.edu
‡Department of Mathematics and Physics, North Carolina Central University, Durham, NC 27707, USA.
Email: atokuta@nccu.edu
§Department of Computer Science and Engineering, Hanyang University, Ansan, South Korea.
Email: hkoh@hanyang.ac.kr

***Abstract*—Location based services (LBSs), which are useful applications of mobile online social network (mOSN), exploit various geographic properties. Location sharing helps people to share their current locations with designated friends and is one important primitive to construct the LBSs. The recent reports showed that a poorly designed location sharing scheme could easily allow the privacy of users to be violated. Over years, lots of efforts are made to provide a privacy-preserving location sharing, but none of them is satisfactory. To address this issue, we introduce a new location sharing scheme in mOSNs with a strong user privacy protection mechanism such that (a) the user's current location as well as (b) the list of friends who will learn the user's current location will be protected from any unintended entity, while the designated friends in the list will learn the exact location of the user. For this purpose, we introduce a new cryptography primitive called the *functional pseudonym* scheme based on Lagrange polynomial with the public social network IDs of the designated friends. Then, the pseudonym of a user is posted on the server along with the current location of the user. While each user can see every posted messages (pseudonym and location pairs), the actual identify of the originator of each pair can be verified only by designated friends, whose identities are used to compute the pseudonym. Most importantly, unlike any of the existing counterparts, our scheme does not assume neither a trusted server nor pre-established secret among the friends.**

I. Introduction

Recently, smartphone is emerging as one of the most crucial staples in our daily lives as well as for businesses [1]. With the widespread use of smartphone, mobile online social networks (mOSNs), which are using the smartphone as their main platform, are widely adopted for various applications. Using online social networks, people can easily exchange various information such as their thought, knowledge, current status, and location with their friends in a timely manner regardless from their current time and location [2]. Location sharing is one significant building block to implement various kind of location based services (LBSs) applications over mOSNs such as local recommendation service, tracking children, proximity notification among users by using the information [3], [4], [5]. For instance, Foursquare is one of the most popular geosocial service providers, and it allows users to register their current location and share that information with nearby friends. Brightkite and Loopt are also well-known LBSs which exploit the location of each user to push notification messages to his/her friends if they are within a certain area at the same time. Unfortunately, all of those location sharing systems have some level of privacy concerns, i.e. location information exposure. This location privacy violation puts users in unpleasant situations such as unwanted location based advertisements or spams, social reputation or economic damage, be a victim of blackmail, and even stalking or physical violence [6]. Based on our comprehensive survey, we conclude that it is desirable for a privacy-preserving LBS over mOSNs to meet the following four requirements.

(a) **Location Privacy**: the current location of users should not be tracked by any unintended entity, which includes the service provider.
(b) **Spatio-Temporal Relation Privacy**: the list of the friends should be completely hidden from unintended entities.
(c) **Semi-trusted Server**: although a server follows a given protocol correctly, it may try to obtain information as much as possible from LBS system.
(d) **Non Pre-established Secret**: it is impractical to assume that each pair of users have pre-established secret in mOSNs which a user can make relationship with strangers.

Over years, many schemes have been proposed to preserve user privacy for location sharing system. Among others, SmokeScreen [7], SMILEs [3], [8], and MobiShares [9], [10], [11], [12] are representative examples of privacy preserving location sharing scheme. However, they are not suitable for mOSNs from the point of view of the aforementioned requirements. SmokeScreens have the unreasonably strong assumption of a trusted third party or pre-established secret for each pair of users to help the proximity notification among users. SMILEs require the users, who would like to detect the presence of each other, to be in the same geographic location beforehand at least once and share some secrets. That users encounter with each other to share a secret is not suitable for the concept of mOSNs, in which users can make relationship with strangers and share location information. In

MobiShares, the location service providers can figure out a user's spatio-temporal relationship by linking queries from the user. Recently, Li et al. proposed a privacy enhanced location sharing scheme for mOSNs [13]. In this scheme, mOSN server consists of two types of servers: a SNS server which provides typical SNS services and location servers which provides a LBS, which is improper because a service provider manages both servers practically. Also, spatio-temporal relation privacy can be infringed by colluding SNS and location servers.

Contribution of This Paper. In this paper, we propose a new location sharing scheme with both location privacy and spatio-temporal relation privacy. We observe that in the series of previous systems such as [3], [10], [11], [12], [13], the privacy of users can be invaded essentially because the server has the enough knowledge to determine whether a user and its friend are nearby or not. To address this issue, we rather use the service provider merely as an anonymous bulletin board to broadcast the anonymized location message of a user such that only intended friends of the user of the moment can verify the actual identity of the user. To implement such system, we propose a new cryptographic primitive, namely the *functional pseudonym*, which is designed based on Lagrange polynomial, to merge the public identities, e.g. social network user ID, of the user's (temporal) friends into a single value. Then, this value is later added to the anonymized location message. In order to verify the identity of the originator of the anonymized location message, each of the designated friends, whose ID was used to generate the functional pseudonym in the anonymized location message, uses Lagrange interpolation along with its own private information (corresponding to the public ID) to reconstruct a secret and extract the identity and the current location of the originator. In this way, we provide location privacy and spatio-temporal relation privacy at the same time without pre-established secrets among users and without a trusted server.

Organization of This Paper. This paper is organized as follows. Section II represents system model, security model, and problem description that considered in this paper, and Section III introduces related work. And we describe our main contribution, privacy-preserving location Multicasting scheme, in Section IV. Section V provides the evaluation for the proposed scheme. Finally, we represent concluding remarks and suggest future research directions in Section VI.

II. Problem Description

This section describes the system model, security model, and problem statement.The notations used in this paper follow Table 1.

A. System Model

The proximity notification service is well-known and widely used concept which notify users about who is nearby and at what distance [14]. The location sharing in mOSN is similar concept to the proximity notification, but crucial difference exists that the LBS in mOSN can use social information to make a sharing group for location sharing.

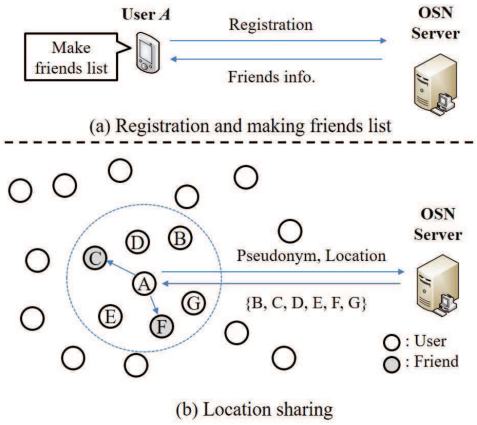

Fig. 1: System Model

Fig. 1 illustrates the system model considered in this paper. According to the location based social networks (LBSNs) classification of [6], our system model can be classified as category I: LBSNs with Exact Location Sharing, and Subtype II: User Authorized Location Sharing that users have the control to choose with whom they share their exact location information. In a nutshell, the system model consists of two entities: a server and users. A service provider provides geosocial services including location sharing through the server, and users can make real time connection with other users and share various information among friends through the server. We define the location sharing as a user allows friends accessing the user's location information, and consider a concept of the service as a sub-application of SNS. Users use the SNS generally, and whenever they want to share location information with friends, they operate the location sharing with a pseudonym as an anonymous identity for privacy preservation.

As shown in Fig. 1 (a), users should be registered to the sub-application first. Each user then makes a designated friends list as a subset of friends list, and only a user in the designated friend list can access to the location information. Fig. 1 (b) shows that the process of finding a nearby friends. If the user uploads the friends list along with location information as meaning of using the location sharing service, the server sends a set of users' information in the user requested range. The user also can broadcast the information to find nearby friends. However, it is more desirable that the user gets a nearby user's information through the server due to some reasons, e.g. loss of signal in building forest, delegation of a computation to the server, hiding an information from non-user, and so on.

While the LBS operates, the user's location privacy as well as spatio-temporal privacy must be protected. Surely, an information of anyone who is not friends must not be revealed.

B. Problem Definition

Given a set of tuples comprised of pseudonyms and location information in the form $\{(P_1, L_1), (P_2, L_2), \ldots, (P_n, L_n)\}$ for a pre-defined range, the problem addressed in this paper

lies in finding a subset of tuples $\{(P_i, L_i)\}_{1 \leq i \leq n}$ generated by friends while preserving two kind of privacy: location privacy and spatio-temporal relation privacy. Also, such a location sharing scheme should be done with no pre-established secret among friends and no trusted server. In general, for anonymous communication with privacy preservation, a user generates a random string and employs it as his/her pseudonym. In this case, a user as well as an attacker are hard to obtain any information from it. However, users necessarily provide some information to make a social relationship and share location information. Therefore, there is a kind of dilemma between providing information and preserving privacy.

C. Adversary and Security Model

In the scope of this work, we consider every unintended entity, who do not have social relation as well as access permission to location information, including a service provider as potential adversaries. Similar as [6], the adversaries in this study has limited capability in which it can only access the publicly available information and is no more than normal users of mOSNs. We assume honest-but-curious model for mOSN server, which follows a given protocol correctly, but may try to obtain information from communication or stored data as much as possible. In addition, we do not consider that the adversaries hack servers to directly access the social relation and location information. Based on this condition, we consider following attacks what can be done by the adversaries.

(a) **Attacks on location information**: Some of previous works show the possibility of location discovery and tracking. In [14], attackers can expose users location information by trilateration attacks. Also, an automated user location tracking system was developed by [6].
(b) **Attacks on pseudonym**: From a given set of pseudonyms, an attacker may try to obtain an information of identity. Also, the attacker may try to distinguish a subset of pseudonym issued by same user. This information can be abused to track users and predict future location. This attack directly link to the location privacy.
(c) **Attacks on spatio-temporal relation**: An attacker has the ability of eavesdropping messages from wired/wireless communication channel. the attacker nearby a certain user observes the user's wired/wireless communication to obtain an information of identity and social relationships. A service provider is also a potential attacker. It can collect log data of downloading users' public identities. By persistent monitoring, the attacker can obtain information about a social relationship among users, furthermore the attacker can infer whole users' list of friends. This attack directly link to the spatio-temporal relation privacy.

From observation of the system and adversary models, following security models are defined:

(a) **Pseudonym indistinguishability**: an attacker cannot distinguish pseudonyms whether it comes from same user or not;

TABLE I: Notations.

Notation	Description
q	k-bit prime number
$\mathbb{Z}_q$	Integers modulo q
$\mathbb{G}$	Cyclic group with prime order q
g	Generator of $\mathbb{G}$
U_i	User i
pk_i, sk_i	Public/private key pair of U_i
PI_i, SI_i	Public/private identity pair of U_i
P_i	Pseudonym of U_i, $P_i = \{p_{i1}, p_{i2}, p_{i3}\}$
L_i	Current location information of U_i
$H(\cdot)$	A collision free hash function that satisfies $\{0,1\}^* \rightarrow \{0,1\}^n$
$h(\cdot)$	A cryptographic one way hash function that satisfies $\{0,1\}^* \rightarrow \mathbb{G}$

(b) **Location privacy**: an attacker cannot invade a user's location privacy regardless of a session. The user's current location information as well as past and future location information cannot be obtained by the attacker. In addition, an attacker who is not a friends of a user cannot obtain both the user's identity and location information.
(c) **Spatio-temporal relation privacy**: an attacker cannot obtain any information about friends from a published pseudonym.

D. Preliminaries

We introduce following two important definitions. They are employed for the security of the proposed scheme.

Definition 1 (Decisional Diffie-Hellman Problem)**.** *Consider a cyclic group $\mathbb{G}$ of order q, and with generator g, the Decisional Diffie-Hellman (DDH) problem [15] is that, given g^a and g^b for uniformly and independently chosen $a, b \in \mathbb{Z}_q$, the value g^{ab} looks like a random element in $\mathbb{G}$. Following two uniform and independent probability distributions are computationally indistinguishable in the security parameter, $n = \log q$:*

(a) g^a, g^b, g^{ab}, where a and b are randomly and independently chosen from $\mathbb{Z}_q$
(b) g^a, g^b, g^c, where a, b, c are randomly and independently chosen from $\mathbb{Z}_q$

Definition 2 (Lagrange Interpolating Polynomial)**.** *The Lagrange interpolating polynomial [16] is the polynomial $f(x)$ of $\tau - 1$ degree that passes through the τ points $(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t)$, and is given by*

$$f(x) = \sum_{\ell=1}^{t} y_\ell \cdot \Delta_{x_\ell, S}(x),$$

where $\Delta_{x_\ell, S}(x)$ is Lagrange coefficient and a set S of elements in $\mathbb{Z}_q$:

$$\Delta_{x_\ell, S}(x) = \prod_{x_m \in S, m \neq \ell} \frac{x - x_m}{x_\ell - x_m}.$$

III. Related Work

Most location based applications need up-to-date user location information to provide better services despite the possibility of user privacy violation [17]. For example, users must disclose their location information to get a location

based service. Users can get more precise services if disclose more information, however user privacy can be even more violated. In order to deal this self-contradicting issue, many scheme were proposed to provide a location information in limited circumstance. In 2007, SmokeScreen was proposed to provide flexible presence-privacy controls for presence-sharing on applications with the identities of co-located users, while user's location information is never revealed without the explicit permission [7]. SmokeScreen also enables presence-sharing among trusted social relationships, as well as untrusted strangers, through trusted brokers which coordinate anonymous communication between them.

MobiShare [9] is a middleware service for mobile wireless terminals and can be used to implement a proximity notification application. However, as the design of MobiShare does not consider privacy and security in mind, the proximity notification system on MobiShare would suffer from various privacy issues. Therefore, MobiShare+ [10], B-Mobishare [11], and N-Mobishare [12], are introduced to address the privacy issue of MobiShare by utilizing multiple pseudonyms and provide a privacy-preserving proximity notification system for mobile online social network. However, Liu et al. [13] pointed out that even though multiple pseudonyms are used for a user, the location server (the service provider who will notify the user if any of the user's friends are within close proximity) which has the knowledge of the social relation of the user may be able to learn the true identity of the user by linking queries from the same user. Based on this observation, Liu et al. proposed to have multiple location servers so that the degree of knowledge on the user at each server can be lowered. However, if the location servers collude, which are possibly owned by the same service provider, this approach suffers from the same problem that MobiShare+, B-Mobishare, and N-Mobishare suffer.

In 2009, Manweiler et al. proposed SMILE a privacy-preserving "missed-connections" service establishing a connection between users who do not have pre-established social relationship through an untrusted service provider [8]. A trust in the SMILE is based on shared encounters which passively exchanged with nearby peers. However, to share encounter, users must be located in a same place and at a same time at least once. In order to establish a social relationship between users, SMILE has to locate users in a short range of place at the same time. A group of users located in the place shares an $encounterkey$, and later users in the group located in a short range of place again, they can contact each other through server using the $encounterkey$ to exchange messages. However, in case SMILE is applied to the system model addressed in this paper, the following drawbacks can occur:

(a) **Impractical rendezvous:** Assuming that users must meet each other at least once for a social relationship is not proper. If the user wants to share location information with a lot of users, the user has to meet all of users, which is impractical in the addressed system model.

(b) **Encounter inefficiency:** Since the encounter key is allocated based on location and time, each user has to store a lot of encounter keys for future relationship. When the user A tries to make a relationship with other user B in same place later, the user A will be confronted with a difficulty of finding an encounter key which shared with B. After finding it, the user A sends a message that encrypted using the encounter key to the server, than the server broadcasts the encrypted message to A's area. Thus the communication overhead increased linearly depending on a number of users in a certain range of area increased.

In 2013, Mohaien et al. proposed an extended version of SMILE. However, it is also based on encounter and not suitable for our system model due to same reasons as SMILE. Recently, the location based services were extended to geosocial networks [3]. With location-aware capabilities, a geosocial network can offer different types of services, such as location sharing, tracking friends, and local recommendation service. Also, a proximity service was proposed in which alerts the user when any of his friends comes into a certain range of the user [18], [19], [5]. However, the rich functionality comes with increased privacy problem, and this problem includes location, absence, co-location, and identity as a sensitive information [4].

In this application, two different kind of privacy issues exist. The first issue is location privacy, and an approach that an attacker can obtain only location information without identity or vice versa and precise location were considered [19], [5]. The second issue is identity privacy and a quasi-identifier scheme was used to deal with this kind of privacy [20], [21]. Particularly, Mascetti et al. proposed a proximity service with complete privacy [5]. When a proximity service satisfies both location privacy and identity privacy, they said it supports complete privacy. In the scheme, they assume untrusted service providers and curious buddies. For this, Mascetti et al. proposed two new protocols: providing complete privacy with respect to a service provider, and controllable privacy with respect to friends. However, it is not suitable for the system model addressed in this paper because they assume that the user's friends user are pre-determined, while in this paper it is assumed no pre-established trust. In addition, it has the drawback of being hard for the user to obtain precise friends' location.

IV. The Proposed Functional Pseudonym based Location Sharing Scheme

In order to preserve privacy, including location privacy during communication, a randomly generated string as pseudonym can be a simple but effective solution. However, a user as well as a service provider hardly can obtain an information, thus making a social relationship impossible under this circumstance. In order to address this problem, we give a functionality to the pseudonym, namely functional pseudonym, while it still has randomness. Using the pseudonym in the proposed scheme, a user can distinguish whether it was maiden with social relationship with the user or not.

Although pseudonym communication is simple and effective solution to preserve privacy, some attacking schemes, statisti-

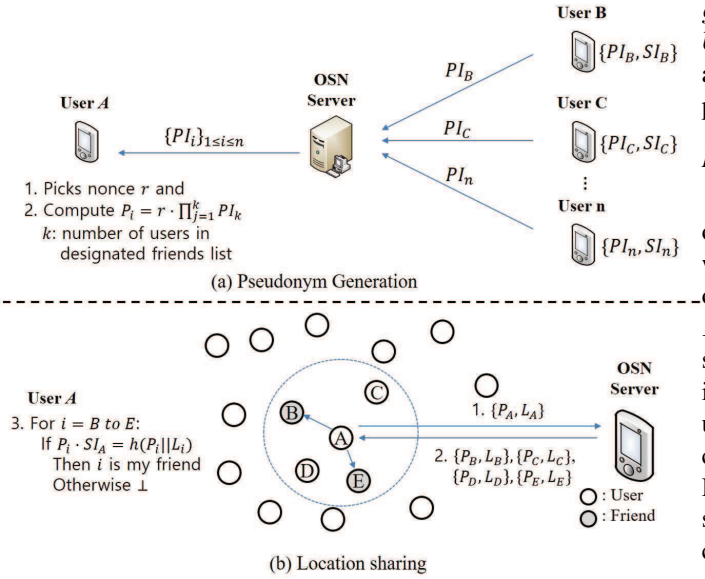

Fig. 2: Proposed Pseudonym Generation and Location Sharing Protocol

cal disclosure and intersection attacks, were introduced [22], [23]. For the robustness against these attacks, we design the functional pseudonym changeable. Whenever a user tries to find nearby friends, the user chooses a new random number which generated by pseudo random generator. We will prove the robustness of changeable functional pseudonym in Section V.

Fig. 2 shows that the flow of the proposed scheme. The proposed scheme consists of four stages. First, when a user joins to the system, it generates a public identity which will be used to make a social relationship among users. Like a general social network service, the user uploads it with other identification information, e.g. pictures, self-introduction messages, and so on. Second, it is assumed a random list download approach to prevent leakage of a social relationship information. Users can make a list of friends safely with this algorithm. Third, it is designed a functional pseudonym. Lagrange interpolation merges a set of public identities into a single value, and the single value has randomness by combining with a random number under Decisional Diffie-Hellman problem. The user uploads it with the current location information to share it with friends. Fourth, another user, who wants to receive location information of nearby friends, sends current location with pseudonym to the server. The server sends a set of (pseudonym, location information) pairs to the user, then the user can find a subset of pairs which issued by friends. In this point, Lagrange interpolation is used to reconstruct pre-computed polynomial.

A. Setup

On input a security parameter 1^k, the setup process first determines a large prime q, $\mathbb{Z}_q$ which represents integers modulo q, and a cyclic group $\mathbb{G}$. Also, it chooses a generator $g \in_R \mathbb{G}$. The global parameters are $\{q, \mathbb{G}, g\}$. Next, each user U_i generates a public/private key pair. It picks $sk_i \in_R \mathbb{Z}_q$ as private key, and computes g^{sk_i} as public key. It outputs public/private key pair (pk_i, sk_i).

B. An Identity

When a user U_i joins to the application, U_i needs to register on the server and it can be represented as public identity (PI) which is used to make a social relationship. At this point, all of users and the service provider can obtain PIs. Also, the PI acts as public key in a cryptosystem, and thus the user U_i stores a related secret identity (SI). At this point, the public identity has property of asymmetric cryptographic key, and the user generates the related secret identity. A general public key cryptosystem such as RSA can be used for public identity. However, it is impossible that put a set of public keys into a single value under such a cryptosystem. Therefore, we design combinable system for users' convenience based on Lagrange polynomial.

The user U_i sends PI and its signature to prevent impersonate attack. Without loss of generality, it is assumed that the PI is stored with other information, such as a photo, to identify friends like real social network applications. Unlike notification services, location sharing systems in mOSN can use social information from server. This information can be used for users to control with whom they share location information.

The user can generate public/secret identity by following process. The user U_i randomly generates $\tau-1$ degree Lagrange basis polynomial $\xi(\cdot)$, picks x_i and computes $y_i = \xi(x_i) \in \mathbb{Z}_q$. Then, a U_i computes $PI_i = g^{y_i \cdot \Delta_{x_i, S_i}(x)}$, where $S_i = \{(x_i, y_i)\}$.

Finally, The U_i uses PI_i as public identity and (x_i, y_i) as secret identity SI_i. Later, the SI_i will be used to confirm a social relationship from a pseudonym. The U_i uploads a signature of PI_i, $\{PI_i\}_{sk_i}$ to the server. If U_i allows to provide location privacy for U_j, U_i adds PI_j on its list of friends.

C. Privacy on the list of friends

In order to add a friend on the list of designated friends, the user U_i searches other user's public identity from the server and adds it on the list of designated friends (Fig. 2 (a)). It is assumed that the user has already the friends list in the server and selects a subset of the list to make the list of friends. In other words, the user U_i has two kind of list: the friends list and the designated friends list, in which the list of designated friends is subset of friends list. The user U_i can download an identity of interested user with dummy identities to muddle the service provider. Adding a user U_j's identity to the user U_i's friends list means that the user U_i allows to share U_i's location information with the user U_j.

In the proposed scheme, a social relationship is represented as the list of friends. The user U_i downloads public identities of interested users from the server to make the list of friends. Since the service provider is a potential attacker, the proposed

scheme has to deal with the privacy of social relationship at this point.

If the user U_i downloads the public identity of a certain user U_j, it means U_i has interest on U_j. Thus, the service provider can roughly infer the list of designated friends of U_i, but no way to assure. For the privacy of social relationship, it is assumed that the user downloads a set of pseudonyms which consists of public identity of U_j and randomly selected users. In addition, it is assumed that the communications between users and the server are anonymized (e.g. with IP and MAC address recycling techniques or Mix Networks [24]) for better privacy.

D. A Pseudonym Generation

Usually a pseudonym is used only for anonymity, however a pseudonym in the proposed scheme has functionality of a friend identification. A user can verify a pseudonym which is generated by one of a friend while hiding the list of friends from a service provider. For this, the proposed scheme generates a pseudonym through three processes.

First, this process generates one more secret to reconstruct secret χ for a friend when s/he checks the pseudonym. A U_i who provides its location to friends randomly generates $\tau - 1$ degree polynomial $\xi(x)$, picks x_t and computes $y_t = \xi(x_t) \in \mathbb{Z}_q$. Then, a U_i computes $PI_t = g^{y_t \cdot \Delta_{x_t, S_t}(x)}$.

Second, a U_i merges public identities on its list of friends using $(2, n)$ threshold secret sharing [16] in reverse. A U_i computes Lagrange polynomial based on a set of public identities $S = \{PI_m\} \bigcup \{PI_t\}$, where m is number of friends on the list:

$$\begin{aligned} PI_t \cdot \prod_{\ell=1}^{m} PI_\ell &= g^{\sum_{\ell=0}^{m} y_\ell \cdot \Delta_{x_\ell, S_\ell}(x) + y_t \cdot \Delta_{x_t, S_t}(x)} \\ &= g^{f(x)}, \end{aligned}$$

where $f(x)$ is Lagrange polynomial and sets $\chi = f(0)$ in which uses as secret in which only friends of U_i can reconstruct it.

Next, pseudonym is generated with computed secrete value χ and Lagrange polynomial $f(x)$. A U_i randomly picks a number $r \in_R \mathbb{Z}_q$. A pseudonym of U_i is $g^{r \cdot y_t \cdot \Delta_{x_t, S_t}(0)}$. Finally, a U_i uploads location information L_i with pseudonym $P_i = \{p_{i1}, p_{i2}, p_{i3}\}$ to the server as follows:

$$P_i = \{g^{r \cdot y_t \cdot \Delta_{x_t, S_t}(0)}, g^r, h(PI_i || L_i) \cdot g^{r \cdot \chi}\},$$

E. Location Multicasting

After completing the designated friends list, the user U_i generates a pseudonym based on that list. If the user U_i wants to find nearby friends, uploads it with his/her current location information. Then, the server sends a set of pseudonyms and location information pairs of nearby users based on U_i's current location. Finally, the user U_i can find nearby friends by checking pseudonyms using the SI (Fig. 2(b)). For the identity privacy, we assume that this process can be done without login.

From received a set of users information L_j, P_j, the user U_i computes following formula to confirm whether it is a friend or not:

$$\begin{aligned} \frac{p_{j3}}{p_{j1} \cdot p_{j2}^{y_i \cdot \Delta_{x_i, S_i}(0)}} &= \frac{h(PI_j || L_j) \cdot g^{r \cdot \chi}}{g^{r \cdot y_t \cdot \Delta_{x_t, S_t}(0)} g^{r \cdot y_i \cdot \Delta_{x_i, S_i}(0)}} \\ &= \frac{h(PI_j || L_j) \cdot g^{r \cdot \chi}}{g^{r \cdot \chi}} \\ &= h(PI_j || L_j), \end{aligned}$$

where $S_i = \{(x_i, y_i)\}$. If the equation holds, the user U_j who is a friend of U_i is on the location L_j, otherwise $\perp$.

F. A Pseudonym Update

Whenever the user U_i adds or removes friends from list, it has to update pseudonyms. In general, a U_i re-computes all the secret sharing process again for update. However, for the efficiency reason, the secret χ is updated by using property of Lagrange polynomial. In case of addition, from a public identity $PI_\alpha = g^{y_\alpha \cdot \Delta_{x_\alpha, S_\alpha}}$ of a new friend, the user U_i computes following equation to update new Lagrange polynomial $f'(x)$:

$$\begin{aligned} g^{f'(x)} &= g^{f(x)} \cdot PI_\alpha, \\ &= g^{f(x) + y_\alpha \cdot \Delta_{x_\alpha, S_\alpha}(x)}. \end{aligned}$$

Then, $\chi' = f'(0)$ becomes new secret. A U_i picks new random number $r' \in_R \mathbb{Z}_q$. A U_i completes pseudonym as follows:

$$P_i = \{g^{r' \cdot y_t \cdot \Delta_{x_t, S_t}(0)}, g^{r'}, h(PI_i || L_i) \cdot g^{r \cdot \chi'}\}.$$

Finally, a U_i uploads L_i', P_i' to the server. Although the pseudonym of U_i was changed, other user and attacker cannot realize the change due to the random number.

On the other hand, in case if U_i removes a U_α from the list of friends, it has to perform whole process of pseudonym generation. For the efficient removing process, we assume that users maintain a list of friends' public identity rather than download it again whenever change occurred. If the user U_i generates a pseudonym with a new random number, a new temporal public identity, and a new secret χ', then the previous friend U_α cannot obtain the location information of U_i anymore.

V. Evaluation

In this Section, we evaluate our scheme in two folds: security and efficiency.

A. Security

The aim of the proposed scheme lies in proximity notification services while preserving two kind of privacy: location privacy, spatio-temporal relation privacy. To evaluate the proposed scheme, security and privacy are proved in three folds. First, a pseudonym is designed with functionality, we should prove the pseudonym has indistinguishability. Second, for security, it is verified if anyone who does not exist on the list of friends cannot obtain the originator's identity. Third, for the spatio-temporal relation privacy, it is proven if an

attacker can to obtain a friend relationship from a given set of pseudonyms. And then, we describe the efficiency evaluation of the proposed scheme. Since the series of SMILE and MobiShare are the most relevant solution among the related works, a comparison between the two schemes is provided in terms of characteristics overheads.

For the security of our proposed scheme, we define the indistinguishability of pseudonyms [25].

Definition 3 (Polynomial-time indistinguishability)**.** *Suppose there exist PRGs that have robustness against polynomial-size circuits. Then, a random number made using the PRG has polynomial-time indistinguishability.*

Two random numbers $X \stackrel{def}{=} \{X_n\}_{n\in\mathbb{N}}$ *and* $Y \stackrel{def}{=} \{Y_n\}_{n\in\mathbb{N}}$*, which are uniformly distributed over* $\{0,1\}^n$ *by the PRG, are indistinguishable in polynomial time if for every probabilistic polynomial-time algorithm* D*, every positive polynomial* $p(\cdot)$*, and all sufficiently large* n*'s,*

$$|Pr[D(X_n, 1^n) = 1] - Pr[D(Y_n, 1^n) = 1]| < \frac{1}{p(n)}.$$

Based on the Definition 3, we prove the indistinguishability of functional pseudonyms, which is our main contribution.

Theorem 1 (Indistinguishability of functional pseudonyms)**.** *Suppose there exist pseudo random generators (PRGs) that have robustness against polynomial-size circuits. Then, a functional pseudonym made with a random number which generated using the PRG has polynomial-time indistinguishability under DDH assumption.*

Proof. Suppose a set of pseudonyms $P = \{P_1, P_2, ...P_m\}$ were obtained by an attacker. We are going to prove any of two given pseudonyms, P_i, P_j from P, are indistinguishable while providing functionality.

A p_{i1} (where $p_{i1} = g^{r'\cdot y_t \cdot \Delta_{x_t,S_t}(0)}$) can be simply written as $g^{r_i\cdot\gamma_i}$, where r_i is a randomly chosen number in $\mathbb{Z}_q$ and γ_i denoted as secret part. From p_{i1} and p_{j1}, each of them can be also represented as $\widehat{g}^{r_i}, \overline{g}^{r_j}$. Since g is a generator of cyclic group $\mathbb{G}$, $\widehat{g}$ and $\overline{g}$ are also generator of $\mathbb{G}$.

In this point, two random numbers r_i and r_j are generated using the PRG, thus it has polynomial-time indistinguishability by Definition 3. Therefore, p_{i1} and p_{j1} are indistinguishable under DDH assumption. □

In the proposed scheme, users can easily generate a new pseudonyms by changing a random number, and it gives no effect on the functionality of pseudonyms. Users can use a new pseudonym whenever they need to communicate with the OSN server. Therefore, attackers are hard to obtain an information from pseudonyms in polynomial-time. Also, it is hard to obtain a correlation between functional pseudonyms even with a statistical attack due to the indistinguishability.

In addition, an attacker may try to infer an information from pseudonyms and public identities, which are public value and every user in the system can access to that values. Thus, we are going to prove the semantic security of the proposed scheme, in which the attacker cannot infer an information from pseudonyms in polynomial time.

Next, we prove that the security of the functional pseudonym.

Theorem 2 (Security of Pseudonyms)**.** *Anyone who is not a friend of* U_i*, he/she cannot notice an originator of* P_i*.*

Proof. To verify this, we refer a security analysis of the secret sharing scheme, which uses $Vandermonde\ matrix$ [26].

The secret χ of a pseudonym, which derived by $(2, n)$ secret sharing can be represented as $f(x) = a(i,0) + a(i,1)x \in \mathbb{Z}_q$, and $f(0) = \chi$. The solution for recover χ can be described by multiplication of the following matrices:

$$\begin{bmatrix} \chi_1 \\ \chi_2 \end{bmatrix} = \begin{bmatrix} 1 & y_i \\ 1 & y_t \end{bmatrix} \times \begin{bmatrix} \sum_{i=1}^{n} a(i,0) \\ \sum_{i=1}^{n} a(i,1) \end{bmatrix}$$

The second matrix of the equation is well-known $Vandermonde\ matrix$ which the determinant of the matrix is non-zero. Thus, the coefficients of the matrix $\{\sum_{i=1}^{n} a(i,0), \sum_{i=1}^{n} a(i,1)\}$ have a unique solution over $\mathbb{Z}_q$.

If a user U_k who does not belong to the list of friends of U_i tries to recover the secret χ, it obtains a linear equation: $\chi_1 = \sum_{k=1}^{n} a(k,0) + y_k \cdot \sum_{k=1}^{n} a(k,1) \in \mathbb{Z}_q$, and $\chi_2 = \sum_{k=1}^{n} a(k,0) + y_t \cdot \sum_{k=1}^{n} a(k,1) \in \mathbb{Z}_q$. Since the coefficient matrix of a $Vandermonde\ matrix$ has a unique solution, above it is stated that two equations derive a unique solution such that $\chi' = f'(0)$. Therefore, any user who is not in friends list cannot recover the secret of a functional pseudonym, thus it cannot notice the originator of it. □

By the Theorem 1 and Theorem 2, Polynomial-time indistinguishability of pseudonyms and Security of Pseudonyms, we can argue an attacker cannot obtain any information from pseudonym. Therefore, the proposed scheme preserves location privacy.

For the spatio-temporal relation privacy, we prove the semantic security of functional pseudonym.

Theorem 3 (Semantic security of functional pseudonym)**.** *The functional pseudonym which proposed in this paper has semantic security, which means that no information can be obtained from functional pseudonyms, and thus attackers cannot obtain information from it in polynomial time.*

Proof. The only information that a pseudonym has is a set of public identities. Additionally, an attacker can obtain another set of public identities downloaded by the user. Using this information, no information about a social relationship and a identity can be derived from a pseudonym.

Let $\mathcal{O}$ be an oracle which can solve the proposed scheme in polynomial time. The $\mathcal{O}$ is represented as: $\mathcal{O}(q, g, P_i, PI_j)$, where q is k-bit prime number, g is a generator of cyclic group generated by q, P_i is pseudonym of the user U_i, and PI_j is public identity of the user U_j. It outputs $true$ if $\mathcal{O}$ can determine PI_j belonging to P_i in polynomial time, otherwise $false$. The oracle $\mathcal{O}$ also can be represented as following form: $(q, g, (g^{r\cdot\alpha}, g^r, g^{r\cdot\chi}), g^\beta)$.

TABLE II: Computational cost based on four major operations. m is the number of friends and n is number of users within the area of interest.

	Public identity	Pseudonym generation	Checking friends
Exponential	2	4	n
Multiplication	1	$m+2$	$4 \times n$
Lagrange coefficient	1	1	n
Hash	.	1	n

Now, we prove DDH assumption using the $\mathcal{O}$. With a, b, c which are chosen at random in $\mathbb{Z}_q$, it can be represented for the distinguishability of g^a, g^b, g^{ab} as follow: $\mathcal{O}(q, g, (g^0, g^a, g^{ab}), g^b)$. In addition, for the distinguishability of g^a, g^b, g^c, it can be represented as $\mathcal{O}(q, g, (g^0, g^a, g^b), g^1)$, $\mathcal{O}(q, g, (g^0, g^b, g^c), g^1)$, and $\mathcal{O}(q, g, (g^0, g^a, g^c), g^1)$.

Thus, if the proposed scheme is solvable using $\mathcal{O}$, then the DDH is also solvable using $\mathcal{O}$. However, DDH is well-known difficulty and already proven that it is not a polynomial time solvable problem. Therefore, the problem of distinguishing a social relationship in the proposed scheme is as hard as DDH by a reduction. □

By the Theorem 2 and 3, we make sure that an attacker cannot obtain a relation information from pseudonyms. Therefore, the proposed scheme preserves spatio-temporal relation privacy.

B. Efficiency

Differently from previous schemes, the proposed scheme has higher level privacy, which provides location privacy and spatio-temporal privacy at the same time. Thus, a direct comparison of the efficiency, at the same level, between previous schemes and the one presented in this work is not possible. Instead, we provide computational costs of the proposed scheme, and then provide comparison among proximity notification service at a same point of view as much as we can.

First, we measure the computational cost of three important functions which used in the proposed scheme: public identity generation, pseudonym generation, and checking a user nearby to confirm it is my friend or not. Table III shows the computational cost of the main functions based on major operations; exponential, multiplication, Lagrange coefficient, and hash.

As shown in Table III, the scheme does not need an expensive operation such as bilinear paring, thus it is efficient in terms of computational cost. It has time complexity of $O(c_1 \cdot m)$ for a pseudonym generation, where c_1 is a constant on the computational cost of multiplication, being m the number of friends in the list. Also, it has time complexity of $O(c_2 \cdot u)$ for checking friends, where c_2 is a constant on the computational cost of checking friends, being u the number of users in a given range of area.

Comparing the proposed scheme with the SMILE in terms of storage and communication overhead. SMILE makes a social relationship based on encounter shared among users located in a same place at a same time. Regardless a number of friends, a user in SMILE has to store encounter keys for every visited location and every time period. A number of stored keys pile up according as the using time increases. It also caused a problem of searching encounter shared with target participants. Meanwhile, the proposed scheme needs to store exactly the same number of public identities as friends in list and no need to find shared secrets.

In communication perspective, SMILE needs more communication overhead than the proposed scheme. Suppose n number of users. Whenever a user sends a message, it will broadcast to the region where the user located through the server. Thus, every user in the section has to receive the sent message and check if it comes from friends or not. At this point, the problem of searching encounter occurred again. If k users $(1 \leq k \leq n)$ is sending messages to find friends in the section, the server has to broadcast k messages, and each user has to receive and check k messages. However, in the proposed scheme, a sent message does not influence other users in the section. The proposed scheme requests a set of user information to the server and checks where the friend is. Therefore, the proposed scheme can drastically reduce communication overhead which is burden to mobile devices.

Situations are similar in series of MobiShare systems [10], [11], [12], [13]. Each pair of user need to share a symmetric key, and this decreases key management efficiency. In terms of finding friends request, a user in MobiShare systems need to make a request message for all users friends. When the user has f friends, the user should compute f request messages using the symmetric keys shared with friends and send f messages to service provider.

Table 3 describes summary of comparison result among proximity notification services in terms of characteristics and overheads. Our scheme was designed to provide a proximity notification service which can preserve (a) location privacy and (b) spatio-temporal relation privacy with no pre-established secret, no trusted server, and no encounter. In addition, our scheme does not need to share and manage a secret with friends for proximity notification services.

While the overhead of SMILE and Mobishare schemes is depending on the number of users friends, the overhead of our scheme is depending on the number of users in an interested area. Since a user in our scheme always sends a single request message to search adjacent friends, our scheme is appropriate for a notification proximity service which a user has a numerous friends and less users in an interested area. Our scheme is also possible to control overhead by adjusting a range of interested area.

VI. Concluding Remarks

This paper presented a privacy preserving location Multicasting scheme in geosocial networks. In order to solve the dilemma between user's privacy and providing information to make a social relationship, the proposed scheme designed a functional pseudonym. Using the pseudonym, the user's identity (and the location) as well as the list of his/her friends can

TABLE III: Comparison of characteristics and overheads in terms of users. m is the number of friends and n is number of users within the area of interest.

	SMILEs	Mobishares	Our scheme
Location privacy	O	O	O
Spatio-temporal relation privacy	O	X	O
Encounter based	O	X	X
Search request	m	m	1
Key storage	m	m	1
Friends confirmation	1	1	n
Communication	m+1	m+1	1+n

be protected from unintended parties while each user can send its exact location to the intended friends secretly and privately. The proposed scheme generates a pseudonym based on a set of public identities in a list using Lagrange polynomial for the computational and storage efficiency. The evaluation section shows that the proposed scheme has complete privacy through proof of randomness while providing such a functionality, security and privacy of pseudonyms. Also, the scheme has sufficient efficiency for resource constrict mobile devices.

ACKNOWLEDGEMENT

This work was jointly supported by US National Science Foundation (NSF) No. HRD-1345219. This work was supported in part by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2015-H8501-15-1007) supervised by the IITP (Institute for Information & communications Technology Promotion), and under the ITRC (Information Technology Research Center) support program (IITP-2015-H8501-15-1018) supervised by the IITP. This work was also supported in part by the NRF (National Research Foundation of Korea) grant funded by the Korea government MEST (Ministry of Education, Science and Technology) (No. NRF-2012R1A2A2A01046986).

REFERENCES

[1] J. J. Wu, M. W. Fang, X. F. Zhang, and T. Y. Wang, "Trusted anonymous authentication scheme for trusted network connection in mobile environment," *Journal of Networks*, vol. 7, no. 9, pp. 1341–1348, September 2012.

[2] 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*, vol. 13, no. 18, pp. 1587–1611, December 2013.

[3] D. Quercia, N. Lathia, F. Calabrese, G. D. Lorenzo, and J. Crowcroft, "Recommending social events from mobile phone location data," in *IEEE ICDM10 Conference Proceedings*. IEEE, December 2010, pp. 971 – 976.

[4] C. R. Vicente, D. Freni, C. Bettini, and C. S. Jensen, "Location-related privacy in geo-social networks," *IEEE Internet Computing*, vol. 15, no. 3, pp. 20–27, February 2011.

[5] S. Mascetti, C. Bettini, D. Freni, X. S. Wang, and S. Jajodia, "Privacy in geo-social networks: Proximity notification with untrusted service providers and curious buddies," *The international Journal of Very Large Data Bases*, vol. 20, no. 4, pp. 541–566, August 2011.

[6] M. Li, H. Zhu, Z. Gao, S. Chen, L. yu, S. Hu, and K. Ren, "All your location are belong to us: Breaking mobile social networks for automated user location tracking," in *ACM MobiHoc14 Conference Proceedings*. ACM SIGMOBILE, August 2014, pp. 43–53.

[7] L. P. Cox, A. Dalton, and V. Marupadi, "Smokescreen: Flexible privacy controls for presence-sharing," in *ACM MobiSys07 Conference Proceedings*. ACM SIGCOMM, June 2007, pp. 233–245.

[8] L. C. J. Manweiler, R. Scudellari, "Smile: encounter-based trust for mobile social services," in *ACM CCS 09*. ACM SIGSAC, November 2009, pp. 246–255.

[9] W. Wei, F. Xu, and Q. Li, "Mobishare: Flexible privacy-preserving location sharing in mobile online social networks," in *IEEE INFOCOM12 Conference Proceedings*. IEEE Computer and Communication Society, March 2012, pp. 2616–2620.

[10] J. Li, J. Li, X. Chen, Z. Liu, and C. Jia, "Mobishare+: Security improved system for location sharing in mobile online social networks," in *IEEE INFOCOM12 Conference Proceedings*. IEEE Computer and Communication Society, March 2012, pp. 2616–2620.

[11] N. Shen, K. Yuan, J. Yang, and C. Jia, "B-mobishare: Privacy-preserving location sharing mechanism in mobile online social networks," in *BWCCA 2014 Conference Proceedings*. IEEE, November 2014, pp. 312–316.

[12] Z. Liu, D. Luo, J. Li, X. Chen, and C. Jia, "N-mobishare: New privacy-preserving location-sharing system for mobile online social networks," *International Journal of Computer Mathematics*, vol. Published online, May 2014.

[13] J. Li, H. Yan, Z. Liu, X. Chen, X. Huang, and D. Wong, "Location-sharing systems with enhanced privacy in mobile online social networks," *IEEE Systems Journal*, vol. Published online, pp. 1–10, April 2015.

[14] I. Polakis, G. Argyros, and T. Petsios, "Where's wally?: Precise user discovery attacks in location proximity," in *ACM CCS15 Conference Proceedings*. ACM SIGSAC, October 2015, pp. 817–828.

[15] D. Boneh, "The decision diffie-hellman problem," in *The 3rd Algorithmic Number Theory Conference Proceedings*. Lecture Notes in Computer Science, June 1998, pp. 48–63.

[16] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, November 1979.

[17] K. G. Shin, X. Ju, Z. Chen, and X. Hu, "Privacy protection for users of location-based services," *Annals of telecommunicationsl*, vol. 69, no. 1-2, pp. 47–62, August 2014.

[18] S. Mascetti, C. Bettini, and D. Freni, "Longitude: Centralized privacy-preserving computation of users' proximity," in *VLDB Workshop on Secure Data Management Conference Proceedings*. LNCS, August 2009, pp. 142–157.

[19] S. Mascetti, C. Bettini, D. Freni, X. S. Wang, and S. Jajodia, "Privacy-aware proximity based services," in *MDM09 Conference Proceedings*. IEEE, May 2009, pp. 31–40.

[20] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: Architecture and algorithms," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 1–18, January 2008.

[21] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, "Preventing location-based identity inference in anonymous spatial queries," *IEEE Transactions on Knoledge and Data Engineering*, vol. 19, no. 12, pp. 1719–1733, December 2007.

[22] D. I. Volinsky, E. Syta, and B. Ford, "Hang with your buddies to resist intersection attacks," in *ACM CCS13 Conference Proceedings*. ACM SIGSAC, November 2013, pp. 1153–1166.

[23] G. Danezis and A. Serjantov, "Statistical disclosure or intersection attacks on anonymity systems," in *Information Hiding*. Lecture Notes in Computer Science, May 2004, pp. 293–308.

[24] C. Cornelius, A. kapadia, D. Kotz, D. Peebles, M. Shin, and N. Triandopoulos, "Anonysense: Privacy-aware people-centric sensing," in *ACM MobiSys08 Conference Proceedings*. ACM SIGCOMM, June 2008, pp. 211–224.

[25] O. Goldreich, "The foundations of cryptography, vol. 1, basic tools," *Cambridge University Press*, 2001.

[26] I.-C. Lin and C.-C. Chang, "A (t,n) threshold secret sharing system with efficient identification of cheaters," *Computing and Informatics*, vol. 24, pp. 529–541, August 2005.