Available online at www.sciencedirect.com

ScienceDirect

Computer Communications 31 (2008) 1181–1191

computer communications

www.elsevier.com/locate/comcom

A TTP-free protocol for location privacy in location-based services

Agusti Solanas *, Antoni Martínez-Ballesté

CRISES Research Group, UNESCO Chair in Data Privacy, Rovira i Virgili University, Department of Computer Engineering and Mathematics, Av. Països Catalans 26, E-43007 Tarragona, Catalonia, Spain

Available online 24 January 2008

Abstract

Location-based services (LBS) will be a keystone of the new information society that is founded on the information and communications technologies (ICTs). Mobile devices such as cell phones or laptops have become ubiquitous. They are equipped with a variety of localisation systems that make them proper for making use of the new LBS. Most of the times, these services are provided by a trusted company (e.g. a telecommunications company). However, the massive use of mobile devices pave the way for the creation of ad hoc wireless networks that can be used to exchange information based on locations. When the exchange of location information is done amongst untrusted parties, the privacy of the participants could be in jeopardy. In this paper we present a novel solution that guarantees the privacy of the users of LBS. Our technique is built up of several modules that progressively increase the privacy level of the users. Unlike the existing approaches, our proposal does not rely on a trusted third party (TTP) to anonymise the users and to guarantee their location privacy.

Keywords: Location-based services; Location privacy; Public-key privacy homomorphism

1. Introduction

The information society is mainly founded on the information and communications technologies (ICTs). The members of the Information Society want to obtain the information we need as fast as possible, from everywhere, and at any time. With this goal in mind, our storage and communication methods have evolved e.g. from the classical paper-based mail carried by postmen to the e-mail that fleetingly crosses the world through optical fibre. This achievements are the result of years of research and efforts. Research on ICTs is a keystone of the Europe's 7th Research Framework Programme [22] that will run until 2013. The European Union is aware of the importance of ICTs for the growth and competitiveness of Europe, and plans to invest over €9 billion in research on ICTs. This investment will lead to the development of more efficient techniques for gathering, storing and sharing data.

Location-based services (LBS) will be one of the most important representatives of these new trends and they will become accessible from everywhere in the near future. The great advances achieved on ICTs in the last few years, and the new ones that we envisage, pave the way for the storage and analysis of a huge amount of information. The balance between the right of the society to information and the right of the individuals to privacy is on the line.

LBS allow users to receive highly personalised information. These services can be accessed by using a variety of mobile devices that can utilise a plethora of localisation technologies. Mobile devices have become ubiquitous and services related to the current position of the users are growing fast. Some examples of these LBS are tourist information services, route planners, emergency assistance services, etc. If the queries that a user makes to a location-based server are not securely managed, it could be possible to infer the consumer habits of the user, e.g. a third party could know that a given user likes Chinese food if the user

* Corresponding author. Tel.: +34 977 55 82 70; fax: +34 977 55 97 10.
E-mail addresses: agusti.solanas@urv.cat (A. Solanas), antoni.martinez@urv.cat (A. Martínez-Ballesté).

0140-3664/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2008.01.007

usually makes queries from a Chinese restaurant. Sending their location, users of LBS could put their privacy in jeopardy. Hence, the European Union has determined a number of measures to be taken in order to address the main privacy problems related to electronic communications. These measures are collected in the Directive 2002/58/EC (Directive on privacy and electronic communications [1]). It deals with the relationship between ICTs and the privacy and security of their users. In addition to classical concepts regarding information security (e.g. encryption or digital signatures), the directive elaborates on some issues related to the privacy of the users, specifically those related to LBS. In fact, the location information is directly related to the privacy of the users since it could be used to determine their consumer habits. According to this directive, the LBS provider *"must inform the users or subscribers [...] whether their location data will be transmitted to a third party"*.

1.1. Privacy problems and their solutions

An LBS basically consists of an LBS provider delivering location-based information and a set of users asking for information. Mobile devices have a variety of ways for determining their approximate location. Thus, we assume in this paper that the utilised devices have this capability (i.e., they can determine their longitude and latitude).

In this scenario, a user (u) asks the LBS provider (P) for some information, sending a message like $\{ID_u$, query, long, lat$\}$, and P returns the required information to u (cf. Fig. 1 for a graphical representation of the information exchange). In this message, 'ID_u' is the identifier of u (e.g. phone number, IP address, etc.), '*query*' represents the information desired by the user, and '$long, lat$' indicate the position of u. Upon this request, P seeks the desired information in its database and returns an appropriate answer '*answer*' to u. Note that, if the users send their exact location to an *untrusted LBS provider P*, the latter could misbehave because it can relate the real location '$long, lat$' to the unique identifier ID_u of u. For instance, the next privacy problems can arise:

- P is able to know if the users are in front of certain shops, so it can flood them with undesired advertisements;
- P can track the movements of the users by studying consecutive queries, so it knows where they have been and when;
- P can send the identifiers of the users along with their location to a *spammer*, and the latter can send undesired location-based advertisements to them.

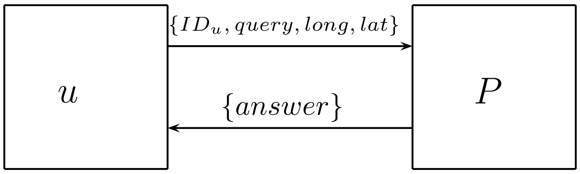

Fig. 1. High-level scheme of the communication between a user and an LBS provider.

In order to avert these possible misbehaviour of the LBS provider, two main solutions have been proposed in the literature:

- *Masking the real positions amongst several users*. By using this technique, inspired in the well-known k-anonymity approach [6,17,16,20], P is not able to distinguish u from a set of k users because they share the same fake/masked location. This indistinguishability makes difficult the tracking and habits inference of the user. Moreover, the ambiguity, associated with the k-anonymity property, makes space for stories [3].
- *Giving an inaccurate position to the LBS provider*. This approach is based on the distortion of the real location. Fig. 2 shows the four quadrants of data utility that we consider. In quadrant ***A*** we place the queries in which the location is too much inaccurate so that the provider cannot obtain any useful information and the user cannot receive a proper answer. In quadrant ***B*** we place the queries with inaccurate locations useless to the provider but good enough to obtain useful information for the user. In quadrant ***C*** we place the queries that obtain proper information from the provider but that their location information is not inaccurate enough to be useless to the provider. Finally, in quadrant ***D*** we consider the queries that are only useful to the provider. This last case takes place when users send accurate locations to the provider and it misbehaves by answering useless information.[1] In this approach, the provided position should be accurate enough so that the information received by u is still useful. However, since the locations collected by P are not exact, they become useless to a spammer (Quadrant ***B***). Note that, although this seems to be a strong assumption, it is reasonable to believe that the user, who really knows the real location, could make a proper use of the information given by the provider. On the contrary, the provider has access to the fake location only.

1.2. Previous work

Most of the previous proposals related to privacy in LBS rely on a trusted third party (TTP) that mediates between users and LBS providers. The so-called *anonymiser* can behave (i) by deleting personal information from the queries of the users before sending them to the LBS providers, or (ii) by hiding the exact position of the user (i.e., modifying it).

[1] In this case the user can put the provider in a blacklist and stop using it.

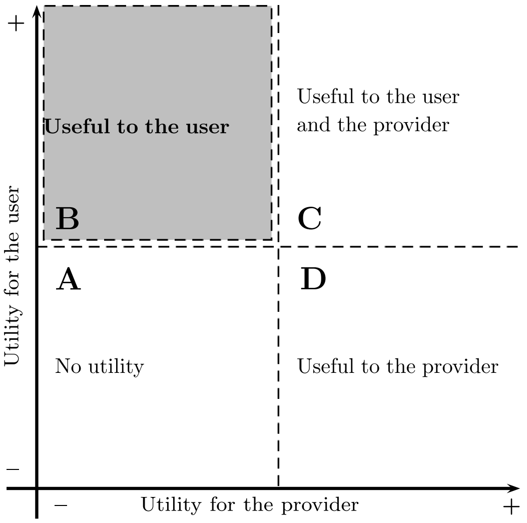

Fig. 2. Quadrants of utility for the user and the provider.

In the first case, a simple protocol is given in Algorithm 1.

In the second case, the anonymiser hides the real location of the user under a *cloaked region* (i.e., a spatial region containing k users) so that each user becomes k-anonymous (i.e., not distinguishable amongst $k-1$ other users). According to [12], the cloaked region must fulfil the property of k-anonymity, but must also consider a spatial cloaking. In that sense, the cloaking algorithm considers a minimum and maximum area sizes and the anonymiser uses the requests from other users (and the location data contained in them) to compute a *masked* location, taking into account the value of k and the area requirements. The masked location can be computed as the centroid of the current locations of the users in the cloaked area. A cloaking protocol scheme is given in Algorithm 2.

In [8], an efficient algorithm for cloaking based on an *anonymity server* is presented. Similar approaches are also presented in [10,13,4]. A slightly different approach is presented in [9]. It consists of a decentralised architecture based on a P2P (peer-to-peer) system. Users utilise P2P techniques to form clusters according to their real location and a certain degree of k-anonymity. In addition, a set of pseudonym servers are used to hide the real identity of the user.

Algorithm 1: Pseudonymisation through an anonymiser

```
1 User $u$ sends a query $\{ID_u, query, long, lat\}$ to the
  anonymiser
2 The anonymiser sends the query $\{pseud_u, query, long, lat\}$,
  where $pseud_u$ is a pseudonym of $u$. The anonymiser keeps
  the real identity $ID_u$ of the pseudonym $pseud_u$ in its
  database
3 The LBS provider answers to the anonymiser
4 The anonymiser sends the answer to the user whose
  pseudonym is $pseud_u$
```

Algorithm 2: k-Anonymity with cloaking

```
1 The anonymiser keeps privacy profiles of its users.
  These profiles may specify, for every day and hour, the
  number of users which take part in the hiding process
  and the maximum and minimum area for the cloaked
  region
2 $u$ Requests a $k$-anonymous location to the anonymiser,
  according to her/his privacy profile
3 The anonymiser uses the requests from other users (and
  the location data contained in them) to compute a
  masked location, taking into account the value of $k$ and
  the area requirements. The masked location can be
  computed as the centroid of the current locations of the
  users in the cloaked area
4 The masked location is returned to $u$
5 $u$ Can use this masked location to query $P$, and she/he
  maintains her/his real location hidden
```

In [5], the authors provide a clear and interesting work on IP location. Specifically, they tackle the problem of privacy in IP location in chapter 10. They summarise the concept of privacy applied to this field. In addition, they point out the possibility of using privacy rules, stated by users, to protect their privacy. How to manage these rules and how to define them are open problems that are currently addressed by Geopriv [21]. Geopriv is a standard for the transmission of location information over the Internet (cf. [2,18]).

The existing proposals for privacy in LBS have some important drawbacks:

- The anonymiser of Algorithm 1 stores the identifiers of its users. Moreover, it can store the location information contained in the queries so the location of its users can be tracked or disclosed to third parties. Users must trust the anonymiser completely.
- The anonymiser of Algorithm 2 also stores the identification and location information of its users. As in the previous case, the security of the system depends on the behaviour of the anonymiser.
- In the approach based on P2P communication technologies [9], users share their real location prior to computing the k-anonymised location. Note that this approach is a generalisation of the previous ones. In this case, the union of the peers leads to their anonymisation. Due to the fact that the peers plainly share their location information, each user must trust the others. Thus, dishonest users could disclose the location information of their peers.
- The rule-based approach proposed in [5,21] has the problem that users must trust the provider. Users state their privacy rules, but they cannot be sure whether the provider is honest. Thus, this approach has shortcomings similar to the ones of previous proposals.

1.3. Contribution and plan of this paper

Location privacy is paramount in LBS. If the proper actions are not taken, the lack of privacy could put the break to the normal deployment of this technology.

In this paper we present a novel solution for providing location privacy to the users of LBS. Our proposal has the next features:

- It does not rely on trusted third parties such as an anonymiser.
- It is a distributed solution. The k-anonymised locations are obtained by the collaboration of the users in a cloaking region.
- It is a modular solution. Depending on the privacy requirements of the users, they can use different modules.
- When the complete solution is used, it guarantees the privacy location of the users even when malicious collusions exist. The location information of the anonymised users cannot be disclosed by the peers, nor by the LBS provider.

The rest of the paper is organised as follows: Section 2 presents the scenario for our model and an analysis of the security requirements related to the number of users of the system. The kernel of our solution is described in Section 3. A security extension of the system is given in Section 4. Section 5 elaborates on the security analysis of different scenarios, and Section 6 concludes the paper.

2. The scenario for our model

The model on which we base our solution can be defined in terms of its main actors and its network infrastructures:

- An untrusted LBS provider (P): Its main task is to give information to users. P is an untrusted provider, thus, users do not want to share their real location with it.
- An LBS user who wants to be anonymised (u): This user wants to get some location-based information from P, but she/he does not want to share her/his real location with P. Thus, the user has to collaborate with other users in order to get her/his location anonymised.
- A set of k users $U = \{u_1, u_2, \ldots, u_k\}$, where $u \in U$ and the remaining $k-1$ users act as companions. We use the term companion to define a user collaborating with u to anonymise[2] her/his location.

These actors make use of two different kind of networks in order to locate themselves and to interact with the others:

- *A Fixed Network (FN):* We assume that users are able to locate themselves by using a FN infrastructure. An assortment of techniques for obtaining the location of mobile devices have been developed namely propagation time, time difference of arrival, angle of arrival or carrier phase [7].
- *An Ad hoc Wireless Network (AWN):* We assume that a user is able to build an AWN with other users in U or to join an existing one. An AWN is created on the fly when users need to exchange information with other users in order to perform a variety of tasks, e.g. to preserve the anonymity of their locations. Users can enter the AWN at any time and leave equally. Thus, the AWN is generally created by the users who want to obtain anonymity by means of requesting information to other users in their cover range. Once the information is exchanged, the AWN is destroyed. We also assume that the users collaborate with each other when they are requested to do so. See [11] for a survey on cooperation methods for AWN.

We assume that LBS users are able to obtain their location and to find $k-1$ companions in their cover range. Although these assumptions seem to be strong, we believe that they will be a reality in the near future. In some countries (e.g. Japan) there are already mobile phones equipped with positioning systems (i.e., GPS) and ad hoc networking capabilities.

2.1. Defining the security requirements

The security required by LBS users can be mainly defined in terms of the number of companions k, amongst which they want to be hidden, and the minimum desired area l^2 where these companions may be distributed. Note that the protection of the real location grows with k and l. At the same time, the differences between the real location and the masked one also grow and the quality of the provided information could decrease.

The task of defining the values of k and l is not straightforward because they may depend on the population density $d = \text{inhabitants}/\text{Km}^2$ of the environment in which the users are located. It is not likely to be possible to find $k = 10,000$ companions to compute a secure location in an area of 1 Km^2 in the Sahara Desert because its population density is very low. Thus, if users impose an area of 1 Km^2, in a region such as the Sahara Desert, the system may inform them about the number of k companions that are likely to be found. To that end, we use the next equations based on the well-known binomial distribution:

$$P(X = k) = \binom{d}{k} p^k (1-p)^{d-k} \tag{1}$$

$$s = P(X \geqslant k) = \sum_{k'=k}^{d} \binom{d}{k'} p^{k'} (1-p)^{d-k'}, \tag{2}$$

where $p = (l^2/L^2)$, being L and l the edges of the considered areas (See Fig. 3).

If d is large enough (i.e., bigger than 30 inh/Km^2) we can easily obtain the same result by applying a Gaussian distribution as shown in the next equation:

[2] We use the term anonymise to be consistent with other approaches, but it could be understood as "obfuscate".

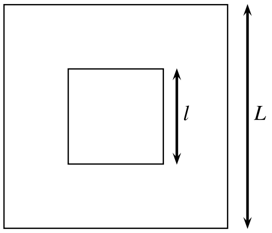

Fig. 3. Representation of l and L.

$$s = P(X \geqslant k) \text{ where } X \sim N(dp, \sqrt{dp(1-p)}) \tag{3}$$

Without loss of generality, we take $L = 1$ Km. Thus, users can define their anonymity area with l and we can determine the number of k companions that are likely to be found with a given probability s. This probability can be set up, being a reasonable choice a value of 0.9. Table 1 shows the number k of companions that is expected for different cities.[3]

2.2. Finding the $k - 1$ companions

Once the users define their security requirements i.e., k and l, by using the method formerly described, they must be able to find $k - 1$ companions for hiding their location from P.

Depending on the number of users into their cover range (r) we can face the next situations[4]:

- *There are no users*: In this case the users cannot proceed with the next steps of the method because they cannot find the required amount of companions.
- *There are less than k users*: If we consider $k = 5$, that is the case illustrated in Fig. 4, user u_1 can contact with u_2 and u_3. By using her/his own means, u_1 is unable to contact more than 2 users, thus, she/he asks for the help of users u_2 and u_3. In the example illustrated in Fig. 4, u_3 can contact u_4 and u_5 whilst u_2 cannot help in finding more users. In this case, the total amount of collaborative users is $k = 5$. This procedure of asking for the help of other users to extend the cover range is repeated until the required number of users k is found. If the procedure ends without the required number of companions, the whole process is stopped.
- *There are k users or more*: In this case the goal is rapidly reached because the needed k users are easily found. In the case in which there are more than k users, say k', the procedure continues with a number of companions[5] between k and k'.

Table 1
Expected companions for different cities, dimensions (d) and areas (l^2).

City	d	$l^2 = 0.01$	$l^2 = 0.04$	$l^2 = 0.09$	$l^2 = 0.16$	$l^2 = 0.25$
Brussels	200	0	4	12	25	42
Sydney	345	1	9	24	46	75
Pretoria	856	4	26	66	123	197
Beijing	888	5	28	68	128	205
Atlanta	1221	7	40	97	178	285
New York	10316	90	387	891	1602	2522
Barcelona	15779	141	599	1374	2465	3875
Paris	24783	227	951	2172	3891	6108
Cairo	35420	330	1369	3118	5578	8750

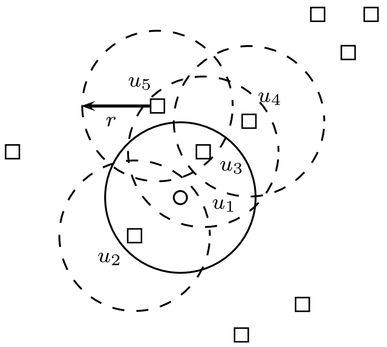

Fig. 4. Illustration of the $k - 1$ companions search.

From now on, we will consider that users are able to find their companions by using some protocol. Although finding companions is essential for our proposal, how these companions are found is not. Thus, for the sake of brevity, we will not discuss possible implementations of such protocols, and we leave this task for future refinements of our proposal.

3. TTP-free location privacy

In this section we present our proposal for providing LBS users with location privacy without using a TTP to anonymise them. A preliminary version of this approach was presented in [19].

Our proposal is based on the computation of a centroid amongst u and her/his $k - 1$ companions so that (i) the position given to the LBS provider is inaccurate but useful enough and (ii) the k users may use the same centroid with the LBS provider so that they become k-anonymous. The scheme of the privacy location protocol without TTP is given in Algorithm 3. As we have explained in previous sections, we assume that LBS users are able to interact with $k - 1$ companions (Step 1 of Algorithm 3). In the next sections we discuss how users can collaborate to compute a common centroid $\overline{U}$ in order to achieve anonymity. We progressively explain the proposed solution for exchanging location information securely. It is introduced step by step, from the simplest version to the most complex and robust one (cf. Fig. 5).

[3] The information on the density of the cities has been obtained from Wikipedia (i.e., http://www.wikipedia.org/).

[4] Without loss of generality we consider that the cover range of all users is r.

[5] When the number of available companions is higher than k, the user is able to select a number of companions between the desired k and the maximum k'. In most cases, the choice of k is the cheapest option in terms of computational cost, and communication resources.

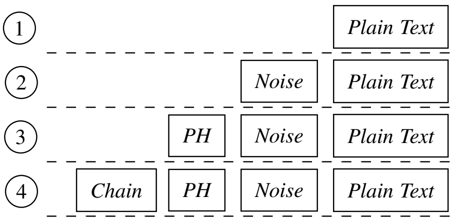

Fig. 5. Scheme of the modules of the protocol.

Algorithm 3: Our basic protocol scheme

```
$u$ Finds $k-1$ companions under some area constraints
For each companion $u_i$
    $u$ Requests the location information to $u_i$
    $u_i$ Sends the information to $u$
$u$ Computes the centroid $\overline{U}$
$u$ Sends the masked location $\overline{U}$ to the LBS provider and to her/his companions
```

Initially, we introduce the simplest scheme without masking (step 1 in Fig. 5). Next, we add some Gaussian noise to the location (step 2 in Fig. 5). After that, we add a privacy homomorphism (PH) to the solution (step 3 in Fig. 5), and finally in Section 4 we complete the scheme with a novel message-sending chain (step 4 in Fig. 5).

3.1. Computing the centroid

Once the user u has received the location information of the companions, u computes the centroid $\overline{U}$ by using Eq. (4),

$$\overline{U} = \left(\frac{\sum_{i=1}^{k} x_i}{k}, \frac{\sum_{i=1}^{k} y_i}{k} \right) \tag{4}$$

where (x_i, y_i) is the location of each user in U. After computing the centroid $\overline{U}$, u sends it to all the companions and to the LBS provider. Using $\overline{U}$, u is identified whilst the real position remains hidden. Note that, as we assume that the companions U are in the same cloaking region, the centroid is accurate enough to let u obtain useful information from P. Moreover, if k and l^2 are properly chosen, the untrusted provider P will not be able to get any useful information (*cf.* Fig. 2, Quadrant B).

This approach, depicted in Fig. 6, is easy and computationally cheap. However, it cannot be really applied because all the messages are sent to u in plain text, thus, u knows the exact location of all the companions. In this case, if u is a malicious user, the location of all the companions is in jeopardy.

This approach has the problem that all the companions must trust u because they send their real locations. Although there are real scenarios in which this situation is possible, (e.g. all the companions are in a list of friends) in many situations users may prefer to hide their real location from the others.

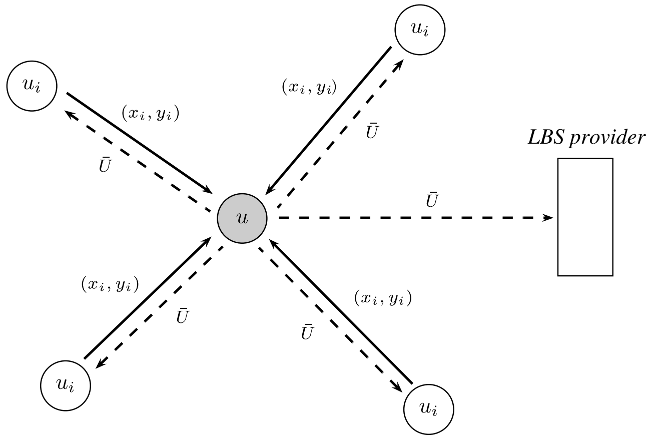

Fig. 6. Plain communication scheme between users in U and P.

3.2. Masking the locations

In order to prevent u from knowing the location of the companions, we extend the previous centroid computation scheme by the addition of Gaussian noise with null average $\sim N(0, \sigma)$. By using a Gaussian pseudo-random numbers generator, each companion u_i can obtain a pair of numbers N_i^x and N_i^y following the desired distribution. Then, these values are added to the real location as Eq. (5) shows:

$$(\hat{x}_i, \hat{y}_i) = (x_i, y_i) + (N_i^x, N_i^y) \tag{5}$$

Once the real locations of the users are masked, they can be freely sent to u in order to let the user compute the centroid $\overline{U}$ (see Fig. 7 for a graphical scheme).

$$\begin{aligned}
\overline{U} &= \left(\frac{\sum_{i=1}^{k} \hat{x}_i}{k}, \frac{\sum_{i=1}^{k} \hat{y}_i}{k} \right) \\
&= \left(\frac{\sum_{i=1}^{k} (x_i + N_i^x)}{k}, \frac{\sum_{i=1}^{k} (y_i + N_i^y)}{k} \right) \\
&= \left(\frac{\sum_{i=1}^{k} x_i}{k}, \frac{\sum_{i=1}^{i=k} y_i}{k} \right) + \left(\frac{\sum_{i=1}^{k} N_i^x}{k}, \frac{\sum_{i=1}^{k} N_i^y}{k} \right) \\
&= \left(\frac{\sum_{i=1}^{k} x_i}{k}, \frac{\sum_{i=1}^{k} y_i}{k} \right) + \left(\overline{N^x}, \overline{N^y} \right)
\end{aligned}$$

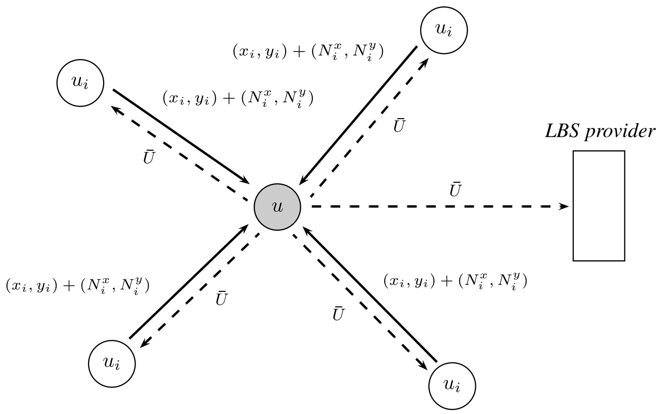
u_i
$(x_i, y_i) + (N_i^x, N_i^y)$
u_i
$(x_i, y_i) + (N_i^x, N_i^y)$
$\bar{U}$
$\bar{U}$
LBS provider
u
$\bar{U}$
$(x_i, y_i) + (N_i^x, N_i^y)$
$(x_i, y_i) + (N_i^x, N_i^y)$
$\bar{U}$
$\bar{U}$
u_i
u_i

Fig. 7. Communication scheme with noise addition.

Finally, u sends the masked location $\overline{U}$ to the LBS provider and to the companions. Note that $\overline{N^x} \approx 0$ and $\overline{N^y} \approx 0$ when k is big enough. Thus, the final centroid is properly computed, although the real locations of the users are masked with noise. Unlike the plain approach, the one described in this section is resilient to a malicious user because she/he does not actually know the location of the other users but the masked ones.

Unfortunately, this approach has a limitation. Due to the use of Gaussian noise with null average, users should not use this technique repeatedly without changing their real location because of the cancellation of the added noise (i.e., the average masked location of these users tends to the real location).

3.3. Location privacy based on a public-key privacy homomorphism

To avoid the limitation of the aforementioned approach, it is necessary to prevent u from knowing the $(\hat{x}_i, \hat{y}_i)$ values of their companions, whilst allowing the computation of a valid centroid $\overline{U}$. To that end, we make use of a public-key privacy homomorphism.

Privacy homomorphisms (PH) were formally introduced in [15] as a tool for processing encrypted data. Basically, they are encryption functions $E_k : T \rightarrow T'$ which allow to perform a set F' of operations on encrypted data without having knowledge of the decryption function D_k. The security gain is obvious because classified data can be encrypted, processed by an unclassified computing facility, and the result decrypted by the classified level.

The PH used must be additive, i.e., one of the operations of F' maps the addition of the unencrypted values. This results in a third party being able to add values but not being able to know which values are being added. In addition, the PH used must be public-key (only the owner of the secret key is able to retrieve the result of the addition) and probabilistic (the encryption algorithm E_k randomly chooses the encrypted value from a set of possible values). For instance, the Okamoto-Uchiyama [14] public-key cryptosystem is probabilistic and has an additive homomorphic property.

Let us assume that there is a Public-key infrastructure (PKI) [23] supplying to LBS providers public keys of a public-key probabilistic and additive privacy homomorphism. This privacy homomorphism has both encryption and decryption functions $E_{pk}(\cdot)$ and $D_{sk}(\cdot)$. The operation mapping the addition of the encrypted values is $\Gamma(\cdot)$. Let P be an untrusted LBS provider whose secret and public keys in the PKI are sk and pk, respectively.

Algorithm 4, depicted in Fig. 8, details all the steps to be taken in order to hide the location of u by means of a public-key privacy homomorphism.[6]

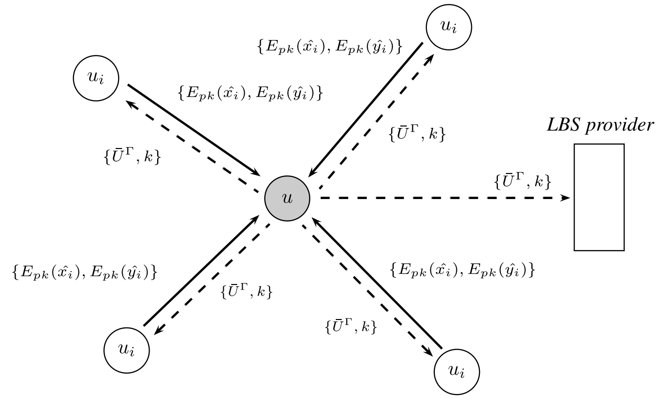
u_i
u_i
u_i
u_i
u
$\{E_{pk}(\hat{x}_i), E_{pk}(\hat{y}_i)\}$
$\{E_{pk}(\hat{x}_i), E_{pk}(\hat{y}_i)\}$
$\{E_{pk}(\hat{x}_i), E_{pk}(\hat{y}_i)\}$
$\{E_{pk}(\hat{x}_i), E_{pk}(\hat{y}_i)\}$
$\{\bar{U}^\Gamma, k\}$
$\{\bar{U}^\Gamma, k\}$
$\{\bar{U}^\Gamma, k\}$
$\{\bar{U}^\Gamma, k\}$
$\{\bar{U}^\Gamma, k\}$
LBS provider

Fig. 8. Communication scheme with a privacy homomorphism.

Algorithm 4: The complete protocol scheme

1 User u initiates a location request with each of the companions $u_i \in U$, sending the message $\{Id_P, u\}$, where Id_P is the identifier of P

2 Upon receiving the message, each companion u_i makes use of the PKI to request the public-key of the LBS provider Id_P to a certification authority

3 By using the PKI, u_i gets pk, signed by a certification authority

4 u_i Checks the validity of pk and sends to u the message $\{E_{pk}(\hat{x}_i), E_{pk}(\hat{y}_i)\}$, where $\hat{x}_i$ and $\hat{y}_i$ is the location of u_i masked according to Expression (5)

5 u Makes use of the PKI to request pk and encrypts her/his masked location

6 u Performs the operation which results in the addition of the unencrypted values, although she/he is not able to see them:

$$\overline{U}^\Gamma = \left(\Gamma_{i=1}^k E_{pk}(\hat{x}_i), \Gamma_{i=1}^k E_{pk}(\hat{y}_i)\right) = \left(E_{pk}\left(\sum_{i=1}^k \hat{x}_i\right), E_{pk}\left(\sum_{i=1}^k \hat{y}_i\right)\right)$$

7 u Sends to P the message $\{\overline{U}^\Gamma, k\}$. u Also sends the message to the companions, so they can use the same message to identify themselves

8 Finally, P decrypts $\overline{U}^\Gamma$, obtains the values $\sum_{i=1}^k \hat{x}_i$ and $\sum_{i=1}^k \hat{y}_i$ and divides them by k to obtain the centroid. Note that P is the only one able to perform this operation because it is the only one who knows the secret key sk

By using our protocol we allow the companions of u to securely send several times their locations to the user, even when they remain in the same location. Assuming that u cannot decrypt the locations sent by the companions, the user cannot see the locations. Moreover, thanks to the probabilistic property of the utilised privacy homomorphism, a malicious user u is not able to track a static companion. Although, this proposal is more secure than the previous ones, it is vulnerable to the collusion of a malicious user and the LBS provider. If u i.e., the user that initiates the protocol, is a malicious user and she/he colludes

[6] Note that certification authorities are widely accepted as a fundamental part of any PKI and they have nothing in common with the concept of TTP related to the anonymisers described in previous sections.

with P, they can share the secret key sk of P and, then, u will be able to decrypt the messages sent by the other users and get their locations.[7] A special case of this situation arises when P and u are the same entity.

4. Chain-based message exchange

Due to the collusion problem enunciated in the previous section, the use of a privacy homomorphism is not secure enough and some more constraints must be added to the protocol.

In this situation we propose to extend the previous solution with the use of a random-order chain, which determines the order in which the messages will be exchanged between the users. With this proposal, the message exchange is no longer centralised by u. On the contrary, the encrypted locations travel along the AWN following a route randomly chosen at every hop.

Algorithm 5 details the main steps of the method. Lines 1, 2 and 3 correspond to the initiation of the hiding protocol by u. Next, from line 4 to 7, the encrypted locations of the k companions travel along the AWN to finally reach the user u (line 8). Once u has the encrypted sum of locations, she/he sends it along with k to P and all her/his companions (lines 9 and 10).

Algorithm 5: The approach extended with PH and a messages chain

```
1 $u$ builds a chain ($C$) consisting of the identifiers of the
  $k-1$ companions that she/he has found
2 $u$ randomly chooses a companion identifier $u_i$ of $C$
3 $u$ sends her/his encrypted location $E_{pk}(x_u), E_{pk}(y_u)$, her/his
  identifier, and $C$ to the randomly selected companion $u_i$
4 while $C \neq \{\o\}$ do
5     The companion $u_i$ uses the properties of the privacy
      homomorphism to add her/his encrypted location
      $E_{pk}(x_{u_i}), E_{pk}(y_{u_i})$ to the encrypted location that she/he
      receives and obtains the encrypted sum of the locations.
      $u_i$ removes her/his identifier from $C$ obtaining a new
      $C \leftarrow (C - u_i)$ and randomly chooses a new identifier $u_j$
      from $C$
6     $u_i$ sends the encrypted result of the summation, $C$ and
      the identifier of $u$ to the next companion $u_j$
7 Once $C = \{\o\}$, the last companion $u_i$ adds her/his
  location to the encrypted sum and sends it back to $u$
8 $u$ sends to $P$ the sum of locations and the number $k$ of
  companions which have taken part into the hiding protocol
9 $P$ decrypts the sum of locations by using her/his secret
  key $sk$ and obtains the centroid $\overline{U}$ by dividing the
  decrypted locations by $k$
```

Fig. 9 shows the scheme of a communication amongst five companions. In step 1 of Fig. 9, the user u starts the communication by creating the list of companions C and sending it along with her/his encrypted location information to the first companion. In step 2 of Fig. 9, the first companion modifies C by removing her/his identifier, adds her/his location to the encrypted location of u and sends all to a randomly selected companion (in this case, the second companion). Next, in step 3, the second companion repeats the procedure and sends the information to the fourth one (note that the companion who receives the message is selected randomly from the list of remaining companions C). In step 4, the procedure is repeated but an empty chain C is sent to the last companion. This last companion repeats the procedure but, as C is empty, she/he ends the message transmission amongst the companions by sending all the encrypted information to the initiator of the protocol i.e., u. Finally, in step 6, the user u sends the aggregated information along with k to P in order the obtain the required information. Although it is not depicted in Fig. 9, u also sends the encrypted summation of locations and k to all her/his companions. Thus, a companion will be able to use the summation latter if her/his position has not changed considerably.

The goal of the addition of a random-order chain in the message-sending is the avoidance of a centralised collector node. Moreover, by allowing a distributed computation of the summation of locations, the protocol becomes more resilient against collusion attacks.

5. Scenarios analysis

In this section we summarise how the different proposed methods perform on a given set of scenarios.

5.1. No attacker scenario

Although this scenario seems to be unrealistic, it is quite common for a user to have lists of friends in the mobile phone, PDA or laptop. Moreover, it is also common to build social networks between friends.

In this special case, using the plain sending of locations is enough to guarantee the location privacy of the user through the most simple k-hiding protocol proposed (*plain sending of locations*).

5.2. Scenario with attackers without collusions

When we face a scenario in which we do not trust the companions, we cannot use the plain scheme proposed in Section 3.1. In this case we must differentiate between two situations, i.e., the user who wants to achieve anonymity is dynamic, or static.

[7] Note that the locations obtained by the malicious user are masked locations. Thus, the privacy of the static users (i.e., the ones that remain in the same location) could be in jeopardy if they repeat the protocol several times. This is not the case for dynamic users.

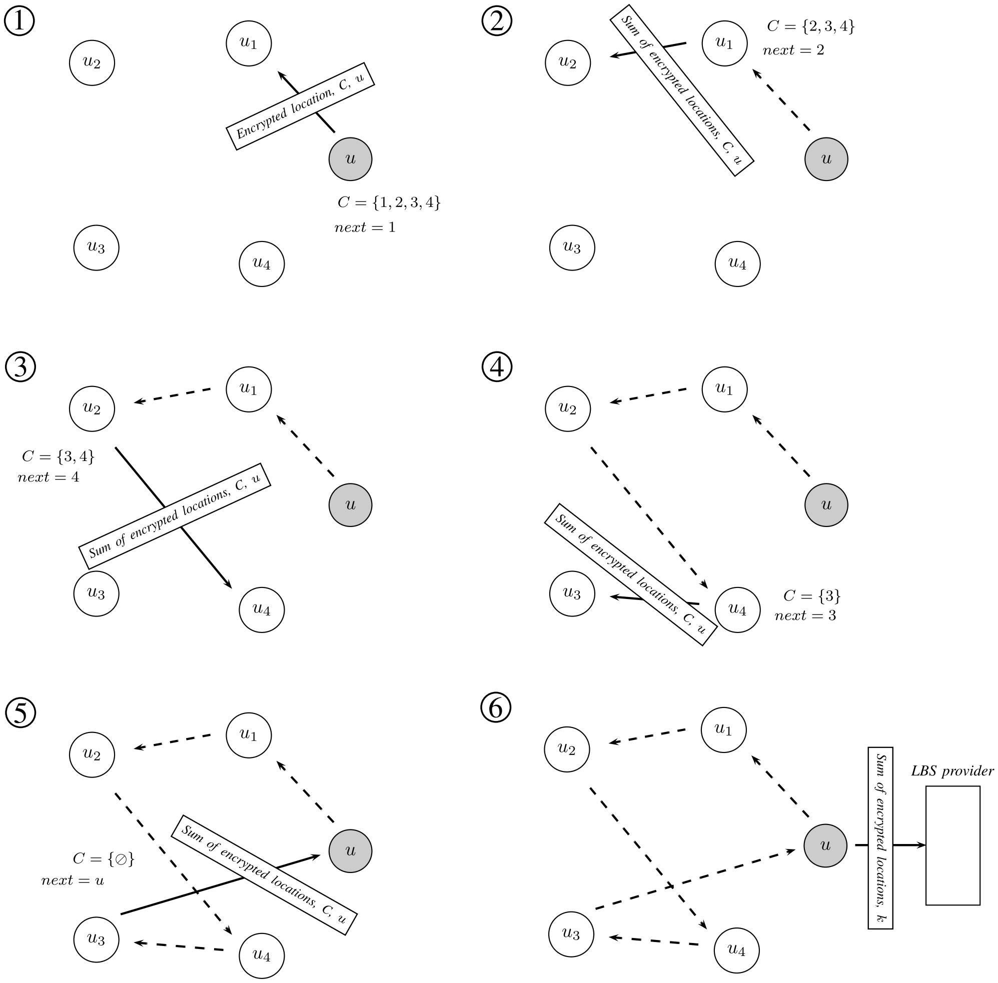

Fig. 9. Communication scheme with a PH and a chain-based message exchange.

- *The user is dynamic.* In this scenario, the user dynamically modifies her/his location, i.e., she/he is moving. In this case, the addition of random noise to the real location is enough to protect the location of the user (*masked locations*).
- *The user is static.* In this case, the user does not change her/his location dynamically, i.e., she/he remains mostly in the same place. As we have shown in Section 3.2, this will put the location of the user in jeopardy due to the noise cancellation. Thus, it is necessary to use the solution based on a privacy homomorphism. The use of such a homomorphism guarantees the location privacy of the user in an scenario without collusions because the users cannot decrypt the encrypted locations which they exchange.

5.3. Collusion scenarios

In this scenario, a malicious user colludes with the location based service provider P. Then, the malicious user knows the secret key of P and she/he can decrypt the encrypted messages sent by the other companions.

Using the approach described in Section 3.3 is not enough to protect the location privacy of the users because, all locations are sent encrypted to u, and due to the collusion between u and P, u is able to decrypt the locations freely.

Due to this fact, the chain-based approach described in Section 4 must be used. In this approach all the messages are not sent to u, instead, they travel through a random chain built on the fly by all the companions. We can face three main situations:

(i) *The colluded user is the first one in the chain*: In this case the user cannot see any location because she/he only receive the summation of all the locations. Thus, she/he cannot infer the individual location of each user.

(ii) *The colluded user is the second in the chain*: In this case the malicious user can obtain the location information of the first user. However due to the addition of random noise, the malicious user cannot exactly determine the location of the first user. The problem is that the user should be in movement in order to avoid the cancellation of noise. Nevertheless, an easy

Table 2
Summary of the robustness of each combination of modules in our solution

Approach	No malicious users	*u* malicious	*u* colluded with *P*
Plain	√		
Masking	√	√	√
Plain + PH	√	√	
Masking+PH	√	√	√
Plain+PH+Chain	√	√	√
Masking+PH+Chain	√	√	√

It is assumed that the users are in movement.

Table 3
Summary of the robustness of each combination of modules in our solution

Approach	No malicious users	*u* malicious	*u* colluded with *P*
Plain	√		
Masking	√		
Plain + PH	√	√	
Masking+PH	√	√	
Plain+PH+Chain	√	√	√
Masking+PH+Chain	√	√	√

The users can remain static, in the same position (no movement is supposed).

solution to this shortcoming consists in the addition of a big random number R to the location of the first user i.e., $x_a + R, y_a + R$. As the privacy homomorphism is additive, the first user can add this noise R and in the end of the protocol,[8] she/he can subtract R from the final summation.

(iii) *The colluded user is not the first neither the second*: In this case, the user will always receive a summation of, at least, two locations. Thus, she/he is not able to determine the locations of any of the other companions.

Table 2 shows the situations in which the different combinations of the modules of our solution are resilient. Note that in Table 2 we assume that the user is in movement so that the cancellation of noise does not affect him/her. Table 3 provides the same information than Table 2 but in this case, the user can remain static i.e., she/he can perform several queries without changing her/his location.

After observing these results, it becomes clear that the only method that provides location privacy in all the scenarios is the one that uses random chains.

6. Conclusions and future work

Location-based services will become a cornerstone in the development of the new Information Society. However, if the LBS are not properly managed, the privacy of the LBS users could be in jeopardy, and that can put the break to the real deployment of these services.

In this paper we have proposed a novel modular solution for providing location privacy to the users of LBS. Our method improves on the existing ones in that:

- It does not rely on the use of a trusted third party (e.g. an anonymiser);
- It is not centralised but distributed;
- It is robust against the collusion of a malicious user and a service provider;
- Due to its modularity, users can utilise the modules that they really need depending on their current needs.
- By using our solution, the users can protect their location privacy even when they are not in movement.

Our protocol could be applied to any LBS in which the exact location of users is not required (e.g. tracking of resources, finding services like pharmacies or restaurants, proximity-based notifications, etc.). Future work includes the study of different collusion scenarios in which the number of malicious users is greater than one. Moreover, we are planing to build a prototype to test the efficiency of our solution with different protocols and technologies.

Disclaimer and Acknowledgements

The authors are solely responsible for the views expressed in this paper, which do not necessarily reflect the position of UNESCO nor commit that organisation. This work was partly supported by the Spanish Ministry of Education through projects TSI2007-65406-C03-01”E-AEGIS” and CONSOLIDER CSD2007-00004”ARES”, and by the Government of Catalonia under grant 2005 SGR 00446.

The authors thank Prof. Dr. Josep Domingo-Ferrer for his insightful comments and suggestions.

References

[1] Directive 2002/58/EC on privacy and electronic communications. Available from: <http://www.europa.eu.int/eur-lex/pri/en/oj/dat/2002/201/20120020731en003700 47.pdf>.

[2] Geographic location/privacy (geopriv). web, September 2006. Available from: <http://www.ietf.org/html.charters/geopriv-charter.html>.

[3] P.M. Aoki, A. Woodruff, Making space for stories: ambiguity in the design of personal communication systems. in: ACM SIGCHI Conf. on Human Factors in Computing Systems, April 2005, Portland, Oregon, USA, pp. 181–190.

[4] R. Cheng, Y. Zhang, E. Bertino, S. Prabhakar, Preserving user location privacy in mobile data management infrastructures. in: 6th Workshop on Privacy Enhancing Technologies (PET), Volume 4258 of Lecture Notes in Computer Science, Springer, Berlin/Heidelberg, 2006, pp. 393– 412.

[8] Note that the first user is also the last one, as we are working in a cyclic chain.

[5] M. Dawson, J. Winterbottom, M. Thompson, IP Location Services, McGraw-Hill, 2006.

[6] J. Domingo-Ferrer, F. Sebe, A. Solanas, A polynomial-time approximation to optimal multivariate microaggregation, Computer and Mathematics with Applications 55 (4) (2008) 714–732.

[7] C. Drane, M. Macnaughtan, C. Scott, Positioning GSM telephones, IEEE Communications Magazine 36 (4) (1998), 46–54,59.

[8] B. Gedik, L. Liu, Protecting location privacy with personalized k-anonymity: Architecture and algorithms, IEEE Transactions on Mobile Computing 7 (1) (2008) 1–18.

[9] G. Ghinita, P. Kalnis, S. Skiadopoulos, PRIVÉ: anonymous location-based queries in distributed mobile systems. in: Proceedings of the Sixteenth International World Wide Web Conference, May 2007, Alberta, Canada, pp. 371–380.

[10] M. Gruteser, D. Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking. in: Proceedings of the First International Conference on Mobile Systems, Applications, and Services, 2003, pp. 31–42.

[11] K. Mandalas, D. Flitzanis, G.F. Marias, P. Georgiadis, A survey of several cooperation enforcement schemes for MANETS, in: Proceedings of the IEEE Int. Symposium on Signal Processing and Information Technology, 2005, pp. 466–471.

[12] M.F. Mokbel, Towards privacy-aware location-based database servers, in: 22nd International Conference on Data Engineering Workshops (ICDEW'06) Atlanta, GA, USA, 2006, pp. 93–103.

[13] M.F. Mokbel, Chi-Yin Chow, Challenges in preserving location privacy in peer-to-peer environments, in: Seventh International Conference on Web-Age Information Management Workshops (WAIM'06), Hong Kong, China, June 2006, pp. 1–8.

[14] T. Okamoto, S. Uchiyama, A new public-key cryptosystem as secure as factoring. in: Advances in Cryptology EUROCRYPT'98, Volume 1403 of Lecture Notes in Computer Science, Springer, Berlin/Heidelberg, 1998, p. 308.

[15] R.L. Rivest, L. Adleman, M.L. Dertouzos, On data banks and privacy homomorphisms, Foundations of Secure Computation (1978) 169–178.

[16] P. Samarati, Protecting respondents identities in microdata release, IEEE Transactions on Knowledge and Data Engineering 13 (6) (2001) 1010–1027.

[17] P. Samarati, L. Sweeney, Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression, Technical report, SRI International, 1998.

[18] H. Schulzrinne, H. Tschofenig, J.B. Morris, J.R. Cuellar, J. Polk, Geolocation Policy: A Document Format for Expressing Privacy Preferences for Location Information, Internet Draft draft-ietf-geopriv-policy, May 2007.

[19] A. Solanas, A. Martínez-Ballesté, Privacy protection in location-based services through a public-key privacy homomorphism. in: Fourth European PKI Workshop: Theory and practice, Lecture Notes in Computer Science, Springer, Berlin/Heidelberg, 2007, pp. 362–368. Palma de Mallorca, Spain.

[20] L. Sweeney, k-anonymity: a model for protecting privacy, International Journal of Uncertainty, Fuzziness and Knowledge Based Systems 10 (5) (2002) 557–570.

[21] H. Tschofenig, H. Schulzrinne, A. Newton, J. Peterson, A. Mankin, The IETF Geopriv and Presence Architecture Focusing on Location Privacy. In Position paper for the W3C workshop on languages for privacy policy negotiation and semantics-driven enforcement, October 2006.

[22] European Union. The Seventh Framework Programme (FP7), 2006. Available from:<http://www.ec.europa.eu/research/fp7/understanding/index.html>.

[23] J.R. Vacca, Public Key Infrastructure, Auerbach, 2004.