Privacy Protection in Location-Based Services Through a Public-Key Privacy Homomorphism

Agusti Solanas and Antoni Martínez-Ballesté

CRISES Research Group
UNESCO Chair in Data Privacy
Dept. Computer Science and Mathematics
Rovira i Virgili University
{agusti.solanas,antoni.martinez}@urv.cat

Abstract. Location-Based Services (LBS) can be accessed from a variety of mobile devices to obtain value added information related to the location of the user. 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. In the latter case, these LBS could be provided by an untrusted party. Sending the location to an untrusted LBS provider could put the privacy of the user in jeopardy. In this paper we propose a novel technique to guarantee the privacy of users of LBS. Our technique consists of several modules, but the highest degree of security is achieved thanks to the use of a public-key privacy homomorphism. Unlike the existing approaches, our proposal does not need any trusted third party to anonymise the users and only makes use of a public-key infrastructure.

Keywords: location privacy, public-key privacy homomorphism.

1 Introduction

Location-Based Services (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 service, router planners, emergency assistance, etc. For a given user of LBS, sending her location could put her privacy in jeopardy.

An LBS basically consists of an LBS provider delivering location-based information and a set of users asking for this 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 the message $\{ID_u, query, long, lat\}$. Upon this request, P seeks the desired information in its database and returns an appropriate answer to u. Note that, if u sends her exact location to an untrusted LBS provider P_u, it can misbehave

J. Lopez, P. Samarati, and J.L. Ferrer (Eds.): EuroPKI 2007, LNCS 4582, pp. 362–368, 2007.
© Springer-Verlag Berlin Heidelberg 2007

because it can relate the real location '$long, lat$' with the unique identifier of u, ID_u, for instance: (i) P_u is able to know if u is in front of certain shops or places, so it can flood her with undesired advertisements; (ii) P_u can track u so it knows where she has been and when; (iii) P_u can send the identifier of u along with her location to a spammer, and the later can send undesired location-based advertisements to the user.

In order to avert these possible misbehaviour of the LBS provider, two main solutions are possible:

- *Hiding the position within other users.* By using this technique inspired in the well-known k-anonymity approach [1,8], P is not able to distinguish u among a set of k users because they share the same fake location. This indistinguishability makes difficult the tracking and habits inference of the user.
- *Giving an inaccurate position to the LBS provider.* The position should be accurate enough so the information received by u is still useful. However, since the locations collected by P are not exact, they become useless to a spammer[1].

To the best of our knowledge, all previous proposals related to privacy in LBS rely on a trusted third party (TTP). One of them is the so-called *anonymizer*, a TTP used for anonymising locations by means of a mediation between users and LBS providers. The *anonymizer* 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).

In the second case, the *anonymizer* 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 among $k-1$ other users). According to [4], the cloaked region must fulfil the requirements 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 *anonymizer* 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.

In [2], an efficient algorithm for cloaking is presented. Similar approaches are also presented in [3] and [6].

1.1 Contribution and Plan of This Paper

In this paper we present a novel location privacy technique based on a Public-Key Infrastructure (PKI) and a public-key privacy homomorphism. Unlike the existing proposals, our technique does not rely on the LBS server acting as a TTP but on the collaboration of the users and an LBS server certified by a certification authority.

[1] Note that, although this seems to be a strong assumption, it is reasonable to believe that the user, who really knows her 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.

Algorithm 1. Our basic protocol scheme

1 u_a Finds $k-1$ companions under some area constraints.
2 For each companion u_i
3 u_a Requests the location information to u_i.
4 u_i Sends the information to u_a.
5 u_a Computes the centroid $\bar{U}$.
6 u_a Sends the masked location $\bar{U}$ to the LBS provider and to her companions.

The rest of the paper is organised as follows. Section 2 describes the simplest variant of our proposal. Section 3 presents an evolution of the simplest version based on a public-key privacy homomorphism and briefly elaborates on its security properties. Finally, Section 4 concludes the paper and points out some future work.

2 TTP-Free Location Privacy

In this section we present the basis of our proposal for providing LBS users with location privacy without using a TTP to anonymise them. The main actors of the model are: (i) An untrusted LBS provider P_u whose main task is to give information to users (since it is untrusted, users do not want to share their real location with it); (ii) An LBS user who wants to be anonymised u_a (she has to collaborate with other users in order to get her location anonymised); (iii) A set of k users $U = \{u_1, u_2, \ldots, u_k\}$ consisting of u_a and $k-1$ companions[2]. We assume that a given LBS user is able to obtain her location and to find $k-1$ companions in her cover range[3]. For the sake of brevity we will not discuss this issue in this paper.

Our proposal is based on the computation of a centroid among u_a and her $k-1$ 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 1.

As we have explained, we assume that LBS users are able to interact with $k-1$ companions (Step 1 of Algorithm 1). In the next sections we discuss how users can collaborate to compute a common centroid $\bar{U}$ in order to achieve anonymity.

2.1 Computing the Centroid

Once the user u_a has received the location information of her companions, she computes the centroid $\bar{U}$ by using Equation 1,

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

[2] We use the term companion to define a user $u_i \in U | u_i \neq u_a$ collaborating with u_a to anonymise her location.

[3] In the case in which the user cannot perform this task, a variety of methods based on the collaboration with other users could be used.

where (x_i, y_i) is the location of each user in U. After computing the centroid $\bar{U}$, u_a sends it to all her companions and to the LBS provider. Using $\bar{U}$, u_a identifies herself whilst her 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_a obtain useful information from P_u.

This approach is easy and computationally cheap. However, it cannot be really applied because all the messages are sent to u_a in plain text, thus, u_a knows the exact location of all her companions. In this case, if u_a 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_a because they send her their real locations. Although there are real scenarios in which this situation is possible, in many situations, users may prefer to hide their real location from the others.

2.2 Masking the Locations

In order to prevent u_a from knowing the location of her 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 Equation 2 shows:

$$(\hat{x}_i, \hat{y}_i) = (x_i, y_i) + (N_i^x, N_i^y) \tag{2}$$

Once the real locations of the users are masked, they can be freely sent to u_a in order to let her compute the centroid $\bar{U}$.

$$\bar{U} = \left(\frac{\sum_{i=1}^{k}(x_i + N_i^x)}{k}, \frac{\sum_{i=1}^{k}(y_i + N_i^y)}{k} \right) = \left(\frac{\sum_{i=1}^{k} x_i}{k}, \frac{\sum_{i=1}^{k} y_i}{k} \right) + \left(\bar{N^x}, \bar{N^y}\right)$$

Finally, u sends her masked location $\bar{U}$ to the LBS provider and to her companions. Note that $\bar{N^x} \approx 0$ and $\bar{N^y} \approx 0$ 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 robust against a malicious user because she 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 the same 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 Location Privacy Based on a Public-Key Privacy Homomorphism

To avoid the limitation of the aforementioned approach, it is necessary to prevent u_a from knowing the $(\hat{x}_i, \hat{y}_i)$ values of their companions, whilst allowing the

computation of a valid centroid $\bar{U}$. To that end, we make use of a public-key privacy homomorphism.

Privacy homomorphisms (PH) were formally introduced in [7] 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[4]. 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 [5] public-key cryptosystem is probabilistic and has an additive homomorphic property.

3.1 Our Proposal

Let us assume that there is a PKI 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_u be an untrusted LBS provider whose secret and public keys in the PKI are sk_{P_u} and pk_{P_u} respectively.

Algorithm 2 details all the steps to be taken in order to hide the location of u_a by means of a public-key privacy homomorphism.

By using our protocol we allow the companions of u_a to securely send several times their locations to her, even when they remain in the same location. Assuming that u_a cannot decrypt the locations sent by her companions, she cannot see the locations. Moreover, thanks to the probabilistic property of the utilised privacy homomorphism, a malicious user u_a is not able to track a static companion. Our protocol is robust against the collusion of u_a and P_u when the companions are in movement.

3.2 Security

We next briefly summarise the security of our proposals. To do so, we check the proposed protocols in two different scenarios with malicious users:

– **Scenario with a malicious user u_a.** Generally, the companions of u_a do not trust her enough to give her their location in plain text. Thus, if they protect their location by adding Gaussian noise with null average, they cannot take part into several anonymity procedures if they do not change their

[4] A privacy homomorphism is additive when one of the operations of F' maps the addition of the unencrypted values.

Algorithm 2. The complete protocol scheme

1 User u_a initiates a location request with each of her companions $u_i \in U$, sending the message $\{Id_{P_u}, u_a\}$, where Id_{P_u} is the identifier of P_u.
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_u} to a certification authority.
3 By using the PKI, u_i gets pk_{P_u}, signed by a certification authority.
4 u_i Checks the validity of pk_{P_u} and sends to u_a the message $\{E_{pk_{P_u}}(\hat{x_i}), E_{pk_{P_u}}(\hat{y_i})\}$, where $\hat{x_i}$ and $\hat{y_i}$ is the location of u_i masked according to Expression 2.
5 u_a Makes use of the PKI to request pk_{P_u} and she encrypts her masked location.
6 u_a Performs the operation which results in the addition of the unencrypted values, although she is not able to see them:

$$\bar{U}^{\Gamma} = \left(\Gamma_{i=1}^{k} E_{pk_{P_u}}(\hat{x_i}), \Gamma_{i=1}^{k} E_{pk_{P_u}}(\hat{y_i})\right) = \left(E_{pk_{P_u}}(\Sigma_{i=1}^{k}\hat{x_i}), E_{pk_{P_u}}(\Sigma_{i=1}^{k}\hat{y_i})\right)$$

7 u_a Sends to P_u the message $\{\bar{U}^{\Gamma}, k\}$. She also sends the message to her companions, so they can use the same message to identify themselves.
8 Finally, P_u decrypts $\bar{U}^{\Gamma}$, obtains the values $\Sigma_{i=1}^{k}\hat{x_i}$ and $\Sigma_{i=1}^{k}\hat{y_i}$ and divides them by k to obtain the centroid. Note that P_u is the only one able to perform this operation because it is the only one who knows sk_{P_u}.

location (because of the cancellation of the added noise). If a given companion is not changing her location during several anonymity procedures she must use the privacy homomorphism proposed in order to prevent u_a from seeing the masked locations and to track her. In this scenario, the plain sending of locations is not secure enough. It is necessary to mask the locations with some Gaussian noise and the companions must be in movement (cf. Section 1). If the companions are static, they have to use the proposed privacy homomorphism to guarantee they location privacy (cf. Section 2).

– **Scenario with a malicious user u_a colluded with P_u.** In this scenario, u_a is colluded with P_u and, thus, she knows the keys pk_{P_u} and sk_{P_u}. A particular case of this scenario is the one in which P_u acts as u_a. Hence, u_a is able to decrypt the messages sent by her companions and she can see the masked locations. In this case, the companions must be in movement in order to guarantee their location privacy. Although the collusion of u_a and P_u is unlikely to happen, if P_u has been found to be colluded with u_a, pk_{P_u} could be revoked and P_u could be put in a blacklist. In this scenario, using a privacy homomorphism is not enough because u_a has access to sk_{P_u}. Thus, it is necessary to be in movement to avert the revelation of the real location due to the noise cancellation.

4 Conclusions and Future Work

In this paper we have proposed a new method for providing the users of LBS with location privacy. Our method is based on a public-key privacy homomorphism

and, unlike the existing proposals, it averts the use of a trusted third party. Our method relies on a basic scheme, consisting in the computation of the average location of a set of k users, and improves it in the sense that it guarantees the location privacy of the users and the location exchange among them by using a public-key privacy homomorphism. Although privacy homomorphisms are widely used in a variety of areas, we believe that a goal of this paper is to propose an algorithm for using them to solve the privacy aspects related to LBS. Future work includes improving the proposed method to allow static users to take part in several anonymity procedures and studying different collusion scenarios, with more than one user colluded with the LBS provider.

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 Government of Catalonia under grants 2002 SGR 00170 and 2005 SGR 00446, by the Spanish Ministry of Education and Science under project SEG2004-04352-C04-01 PROPRIETAS. The authors thank the insightful comments of the reviewers, which helped in improving the article although it has been shortened.

References

1. Domingo-Ferrer, J., Sebé, F., Solanas, A.: A polynomial-time approximation to optimal multivariate microaggregation. In: Computer and Mathematics with Applications (in press, 2007)
2. Gedik, B., Liu, L.: A Customizable k-Anonymity Model for Protecting Location Privacy (2004)
3. Gruteser, M., Grunwald, D.: Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. In: Proceedings of the First International Conference on Mobile Systems, Applications, and Services, pp. 31–42 (2003)
4. Mokbel, M.F.: Towards Privacy-Aware Location-Based Database Servers. International Workshop on Privacy Data Management, PDM (April 2006)
5. Okamoto, T., Uchiyama, S.: A new public-key cryptosystem as secure as factoring. Lecture notes in computer science, pp. 308–318
6. Bertino, E., Cheng, R., Zhang, Y., Prabhakar, S.: Preserving user location privacy in mobile data management infrastructures. In: Proceedings of Privacy Enhancing Technology Workshop (PET) (2006)
7. Rivest, R.L., Adleman, L., Dertouzos, M.L.: On data banks and privacy homomorphisms. Foundations of Secure Computation, 169–178 (1978)
8. Sweeney, L.: k-anonymity: A model for protecting privacy. Fuzziness and Knowledge Based Systems 10(5), 557–570 (2002)