Location Privacy in Chain-Based Protocols for Location-Based Services

Marius O. Gheorghita
UNESCO Chair in Data Privacy
Dept. of Comp. Eng. and Maths
Universitat Rovira i Virgili,
Av. Països Catalans 26,
E-43007 Tarragona,
Catalonia (Spain)

Agusti Solanas
UNESCO Chair in Data Privacy
Dept. of Comp. Eng. and Maths
Universitat Rovira i Virgili,
Av. Països Catalans 26,
E-43007 Tarragona,
Catalonia (Spain)

Jordi Forné
Dept. of Telematics Engineering
Technical University of Catalonia,
Jordi Girona, 1-3,
E-08034 Barcelona,
Catalonia (Spain)

***Abstract*— Mobile communication devices have rapidly proliferated in all developed countries. Most of these devices can determine their location, and their computation and storage capacities have greatly increased in recent years. Therefore, Location-Based Services (LBS) will become cornerstones of the new Information Society. Most of the time, LBS are provided by trusted companies such as telecommunications companies. However, the wide availability of these technologies can foster the appearance of alternative non-trusted providers.**

When location information is exchanged amongst untrusted parties, the privacy of the participants could be in jeopardy. Thus, some methods have been proposed in the literature to provide users with location privacy, and most of them rely on a trusted third party (TTP) that is used to centralise all location information prior to sending it to a non-trusted provider. Unfortunately, in many situations TTPs are not available.

In this paper, we provide a brief overview of the state of the art of LBS privacy, we recall a novel TTP-free proposal and, finally, we analyse its robustness against the attack of several passive malicious users.

I. Introduction

Research on Information and Communication Technologies (ICTs) is a keystone of the Europe's 7th Research Framework Programme [1] that will run until 2013. The European Union (EU) plans to invest over € 9 billion in research on ICTs with the aim to develop more efficient techniques for gathering, storing and sharing data. These efforts will pave the way for the storage and analysis of huge amounts of information. Due to these advances in ICTs, users' privacy can be jeopardised. The balance between the right of the society to information and the right of the individuals to privacy is on the line.

Location-Based Services (LBS) are important representatives of the new ICTs that will be a hot topic, especially because they will become accessible from everywhere and at any time. LBS permit users to receive highly personalised information based on their location. Some examples of LBS are tourist information services, route planners, emergency assistance services, etc. Unfortunately, using LBS is not 100% positive. If the queries that a user submits to a location-based server are not securely managed, the provider of the service can infer the consumer habits of the user, *e.g.* it could be inferred that a given user likes Italian food if the user usually submits queries from an Italian restaurant. Hence, the European Union has pointed out several 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 [2]). The directive deals with the relationship between ICTs and the privacy and security of their users. It elaborates on privacy issues related to LBS. In fact, the location information is directly related to the privacy of the users. According to this directive, the LBS provider *"must inform the users or subscribers* [...] *whether their location data will be transmitted to a third party"*.

LBS have great potential. However, if the security and privacy of the users are neglected, the deployment of LBS can be slowed down or even stopped. Therefore, a wide variety of techniques have been proposed to protect the privacy of LBS users and foster the deployment of these services.

A. Contribution and plan of the paper

In this paper we provide a concise overview of the state of the art of location privacy for LBS. We emphasise that most of the current solutions are centralised and TTP-based, and only a few consider non-centralised TTP-free approaches.

We recall a brand new non-centralised TTP-free protocol (*cf.* [3][4]), and we point out its main advantages. Last but not least, we provide a security statistical analysis of its resiliency against passive collusion attacks.

The rest of the paper is organised as follows. Section II provides an overview of the most relevant solutions to achieve location privacy in LBS proposed in the literature. Section III recalls a novel non-centralised TTP-free method whose security is studied in Section IV. We finish the article with a conclusion in Section V.

II. Background

In LBS we can distinguish two main actors: users and service providers. In the simplest form of communication amongst these actors, a single user directly sends a query

978-0-7695-3188-5/08 $25.00 © 2008 IEEE
DOI 10.1109/ICDT.2008.19

IEEE computer society

along with her/his ID and location to a provider in order to obtain some information in return (*cf.* Figure 1-A). In this naïve case, users send their location to a provider hoping that the provider will properly manage this information. In other words, users trust providers. Although this situation is possible (*e.g.* the provider is a trusted telecom company), in general, users cannot trust providers.

With the aim to overcome this privacy problem many methods have been proposed. Most of them rely on trusted third parties (TTP)s that mediate between users and LBS providers. The so-called *anonymisers* use two different approaches: (i) they can replace personal information from the queries of the users by pseudonyms before sending them to the LBS providers, or (ii) they can hide the exact position of the user. In the first case (*pseudonomysation*), the TTP stores a correspondence table that relates real IDs and pseudonyms. When it receives an answer from the provider, it replaces the pseudonym by the real ID and sends the answer back to the user. It follows a principle very similar to the network address translation (NAT) used in intranets.

In the second case, *anonymisers* hide the real location of users under a *cloaked region* (*i.e.* a region containing, at least, k users) so that each user becomes k-anonymous (*i.e.* not distinguishable amongst $k-1$ other users). This k-anonymity property [5], [6], [7] is leveraged from the Statistical Disclosure Control (SDC) field in which the main goal is to protect the privacy of the respondents whose data are released (mainly by national statistical agencies). In SDC, a data set is said to be k-anonymous for $k > 1$ if, for each combination of key attributes, at least k records exist in the data set sharing that combination. In the special case of LBS privacy, k-anonymity is achieved when there are, at least, k users in a cloaking region so that all they are represented by the same fake location.

According to [8], the cloaked region must fulfil the property of k-anonymity, but must also assure a spatial cloaking. To do so, the cloaking algorithm defines a minimum and maximum area sizes and *anonymisers* use the requests from other users (and the location data contained in them) to compute a *masked* location. This location can be computed as the centroid of the current locations of the users in the cloaked area. In [9], an efficient algorithm for cloaking based on an *anonymity server* is presented. Similar approaches are also presented in [10], [11] and [12].

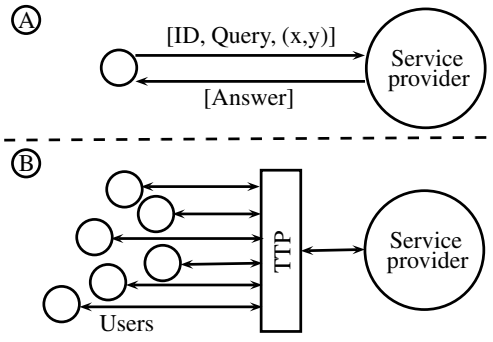


Fig. 1. A: A single user interacts with a service provider directly. B: A set of users uses a mediator (TTP) to interact with a service provider

The approaches described above are centralised in the sense that *anonymisers* receive all the queries from users and, then, they modify the location conveniently. On the contrary, in [13], a decentralised architecture based on a peer-to-peer (P2P) system was proposed. Users use 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 users.

In [14], the authors point out the possibility of using privacy rules, stated by users, to protect their privacy. How to manage and define these rules are open questions that are currently being addressed by Geopriv [15]. Geopriv is a standard for the transmission of location information over the Internet (*cf.* [16],[17]).

All these proposals have some important drawbacks:

- The *anonymiser* based on pseudonyms stores the ID of its users. In addition, it can store their location information that is contained in the queries. By doing this, users can be tracked and their location information can be disclosed to third parties. In conclusion, users must trust *anonymisers* completely and the original problem (*i.e.* the lack of available TTPs) is not solved.
- The *anonymiser* based on cloaking also stores the ID and the location information of its users. As in the previous case, the security of the system depends on the behaviour of *anonymisers* and users must trust them to use the service. Again, the problem is not solved.
- The P2P-based approach requires users to share their real location prior to computing the k-anonymised location. This approach is a generalisation of the previous ones in which the union of the peers leads to their anonymisation. Unfortunately, the peers involved in the anonymisation process, plainly share their location information. Thus, each user must trust the others because dishonest users can disclose the location information of their peers. Again, the problem is not solved.
- The rule-based approach has the problem that users must trust providers. Users state their privacy rules, but they cannot be sure whether the provider respect that rules. Thus, this approach has shortcomings similar to the ones of previous proposals.

It is clear that the original problem is still unsolved, and that there is a need for a protocol that permits users to privately send queries to providers without using TTPs.

We will recall a solution to this problem in the next section.

III. TTP-FREE LOCATION PRIVACY

In this section we recall the main points of the location privacy protection method proposed by Solanas et al. in [3] *(centralised)* and extended in [4] *(non-centralised)*. The main aim of the method is to provide location privacy whilst avoiding TTPs. To do so, location privacy is achieved by means of the collaboration of several (*e.g.* k) users that exchange their

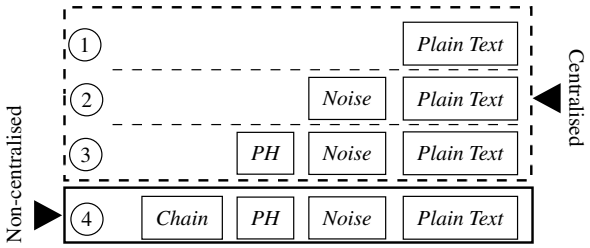


Fig. 2. Scheme of the modules of the protocol.

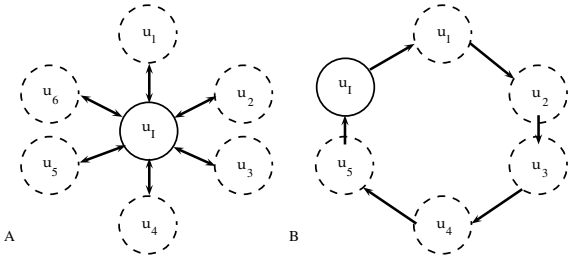


Fig. 3. A: centralised scheme with the Initiator collecting all messages. B: non-centralised chain-based scheme.

location information to compute a centroid $\bar{C}$ that will replace their real location.

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

Once the centroid is computed, users use it as their own location. Thus, service providers cannot distinguish users and they become k-anonymous. The protocol allows users to compute the centroid without jeopardising their location privacy. The first version of the protocol defined in [3] is centralised and has three modules (*cf.* Section III-A). The second version of the protocol defined in [4] extends the protocol by means of a message exchange chain and provides a non-centralised approach (*cf.* Section III-B). Figure 2 shows a scheme of the four layers of the protocol.

A. Centralised approach

This approach assumes that a user U_I initiates the protocol and collects all the location information from the others to compute the centroid (*cf.* Figure 3 A). In the following sections, we briefly summarise how the centroid is computed in each layer.

1) Plain sending of locations: In this first layer, location information is sent in plain text. Once the central user has all the information he/she can compute the centroid by using Equation (1). Unfortunately, it is obvious that sending the locations in plain text offers no privacy to the users (except the central user).

2) Adding noise: The second layer of the protocol adds some Gaussian noise with null average to the real location of the users, so that they obtain a masked location $(\hat{x}_i, \hat{y}_i)$.

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

The central user receives all the masked locations from the other users and compute the centroid by using Equation 3.

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

$$= \left(\frac{\sum_{i=1}^{k} x_i}{k}, \frac{\sum_{i=1}^{k} y_i}{k} \right) + \left(\bar{N^x}, \bar{N^y} \right) \quad (3)$$

Note that $\left(\bar{N^x}, \bar{N^y} \right)$ tends to 0 because the Gaussian noise used has null average. Thus, the centroid is properly computed whilst the location privacy of the users is preserved. Unfortunately, if a user repeats this protocol without changing her/his real location (*i.e.* a static user), the central user can determine her/his real location because of noise cancellation.

3) Using a Privacy Homomorphism: To avoid the limitation of the previous layer, it is necessary to prevent the central user from knowing $(\hat{x_i}, \hat{y_i})$, whilst allowing her/him to compute a valid centroid $\bar{C}$. To that end, a public-key privacy homomorphism is used in the third layer.

Privacy homomorphisms (PH) 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 (*e.g.* the location of the users) can be encrypted, processed by an unclassified computing facility (*e.g.* the central user), and the result decrypted by the classified level (*e.g.* the service provider). The PH used in this layer is the Okamoto-Uchiyama [18], a public-key cryptosystem that is probabilistic and has an additive homomorphic property. 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 $\Gamma(\cdot)$. Where sk and pk are the private and public key of the service provider.

Users encrypt their location by using $E_{pk}(\cdot)$. They send the encrypted locations to the central user and she/he compute the summation using $\Gamma(\cdot)$. Finally, the central user sends the encrypted summation along with the number of participants (k) to the service provider that can decrypt the summation using $D_{sk}(\cdot)$ because she/he knows the private key sk, and can compute the centroid dividing this summation by k.

B. Non-Centralised approach

This approach assumes that a user U_I initiates the protocol but she/he does not collect all the location information from the others. On the contrary, location information travels along a chain created by the users (*cf.* Figure 3 B). This approach provides the users with more privacy because it overcomes the problems associated to the centralised approaches (*e.g.* if the central user is colluded with the service provider, she/he can decrypt the locations sent by the other users using the PH).

Initially, the initiator U_I creates a list L containing the IDs of all the users that agreed to take part in the computation of the centroid. Then she/he send L and her/his encrypted location to a randomly selected member of the list. The user who receives the message, selects a new user from L and

Source	Dest.	Message	Action
User 1	User 2	$(x_1 + R, y_1 + R)$ - [**1**,3,4]	Send location and ist
User 2	User 4	$(x_1 + R + x_2$ $, y_1 + R + y_2)$ - [**1**,3]	Send sum of locations and list
User 4	User 3	$(x_1 + R + x_2 + x_4$ $, y_1 + R + y_2 + y_4)$ - [**1**]	Send sum of location and list
User 3	User 1	$(x_1 + R + x_2 + x_4 + x_3$ $, y_1 + R + y_2 + y_4 + y_3)$	Send sum of location to the initiator
User 1	ALL	$(x_1 + x_2 + x_4 + x_3$ $, y_1 + y_2 + y_4 + y_3)$ - k	Remove R and send the result to all participants

TABLE I

EXAMPLE OF MESSAGES IN THE CHAIN PROTOCOL WITH 4 USERS.

sends her/him L and the encrypted sum of locations (*i.e.* each user adds her/his location to the sum of locations that she/he receives). At the end of the process, the encrypted summation of all locations comes back to the initiator. Then she/he send this encrypted summation and the number of participants k to the service provider, who can decrypt the summation and compute the centroid dividing by k. A little example of this message exchange is given in Table I. For more details see [4].

Although this approach is resilient to the collusion of the provider and a single user, there is no study on its security against the collusion of several users and the provider. This study is provided in the next section.

IV. MODEL AND STATISTICAL SECURITY ANALYSIS

In this section we explain the model that we use to analyse the resiliency of the chain-based protocol to the collusion of several users and the provider.

The main actors of the model are:

- Static users (S): These are users that remain in the same position for several executions of the protocol. Thus, noise added to their locations is cancelled and cannot be used to protect their location privacy.
- Malicious users (M): These are users colluded with the service provider and they can decrypt the information they receive. Thus, using a PH does not help protect the location privacy of users and all the security relies on the summation of user locations.

With the aim of analysing a general chain of n users, we convert the chain into a string (*cf.* Figure 4), we define a binary alphabet $\Sigma = \{M, S\}$ and we consider the set of strings of length n, (Σ^n), contained in Σ^*[1].

Due to the fact that malicious users (M) can decrypt messages, if a static user (S) is located between two malicious users in the chain, her/his location is revealed. Thus, in order to determine the probability that the location of a static user is revealed, we consider the number of patterns Malicious-Static-Malicious (MSM) contained in Σ^n.

[1]Σ^* denotes the set of all finite strings over the alphabet Σ, and $*$ denotes the Kleene star operator.

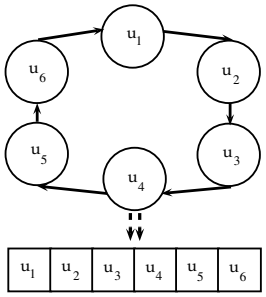


Fig. 4. Converting a chain into a string

1	S S S M M M	S M S S M M	11
2	S S M M M S	M S S M M S	12
3	S M M M S S	M M S S M S	13
4	M M M S S S	S M M S S M	14
5	S S M S M M	S M S M S M	15
6	S M S M M S	M S S S M M	16
7	M S M M S S	M S M S M S	17
8	S S M M S M	M M S S S M	18
9	S M M S M S	M S S M S M	19
10	M M S M S S	M S M S S M	20

Fig. 5. Strings in the set Σ^6 for our example with 3 malicious users and 3 static users. The pattern MSM is marked on each string.

Thus the probability of success is defined as:

$$P(success) = \frac{NCU}{NS} \tag{4}$$

where NCU is the number of compromised users (*i.e.* the number of patters MSM found in Σ^n), and NS is the total number of static users in Σ^n.

The following example helps understand the idea:

Example 1: Given a chain of 6 users ($n = 6$), which is the probability of obtaining the location of a static user if there are 3 malicious users in the chain? See Figure 5 for a graphical representation of all strings and patterns for this particular case.

First we compute $NS = 60$ *i.e.* 20 strings multiplied by 3 S in each string. Then we count how many times the MSM pattern appears. In this case $NCU = 15$. Thus, the probability of revealing the location of a static user is:

$$P(success) = \tfrac{15}{60} = 0.25$$

□

In the previous example, it is easy to count the number of strings, and the number of MSM patterns. From the 15 MSM patterns, 3 are obtained by closing the chain (*e.g.* pattern 13 in Figure 5) and 12 are found in the string. However, when the number of users (n) grows, it is not straightforward to manually find these patterns and a general formula is

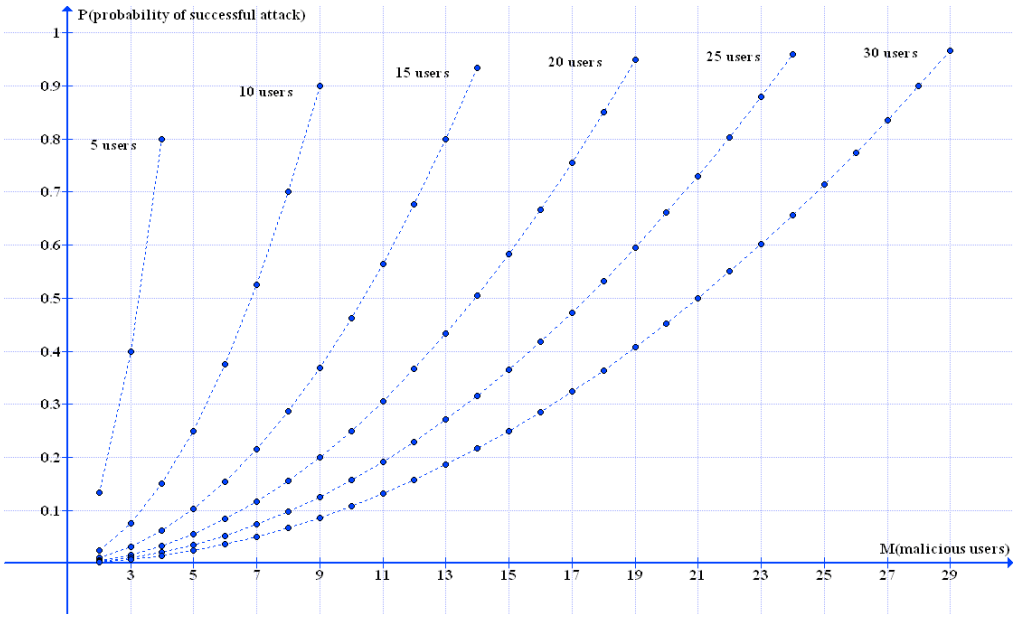


Fig. 6. Probability of revealing a static user location depending on the total number of users (*i.e.* 5, 10, 15, 20, 25 and 30 users) and the number of malicious users (*i.e.* from 3 to 29) in the chain.

necessary. The probability of success can be expressed as:

$$P(success) = \frac{\left[\begin{pmatrix} n-3 \\ m-2 \end{pmatrix} + \begin{pmatrix} n-2 \\ m-2 \end{pmatrix}(n-m)\right]}{\left[\begin{pmatrix} n \\ m \end{pmatrix}(n-m)\right]} \quad (5)$$

where n is the total number of users, and m is the number of malicious users.

The probability of a successful attack is given in Equation (5) as the number of static users compromised, divided by the total number of possible static users exposed.

The total number of static users exposed is the total number of possible strings $\begin{pmatrix} n \\ m \end{pmatrix}$ multiplied by the number of S in each string (*i.e.* $(n-m)$).

The number of compromised users is equal to the number of MSM patterns found in all possible strings. $\begin{pmatrix} n-3 \\ m-2 \end{pmatrix}$ is the number of the patterns formed by closing the chain, and $\begin{pmatrix} n-2 \\ m-2 \end{pmatrix}(n-m)$ is the number of MSM patterns that can be found in the string.

In Table II the probability of a successful attack on a static user is given. Moreover, a graphical representation of the probability of revealing the location of a static user for several values of n and m is provided in Figure 6. It can be observed that in order to have at least a probability of success of $\frac{1}{4}$, the number of malicious users must be $\frac{n}{2}$ of n, or more.

n	Number of malicious users (m)					
	5	6	7	8	9	10
6	0.8333					
7	0.5714	0.8571				
8	0.4166	0.6250	0.8750			
9	0.3174	0.4761	0.6667	0.888		
10	**0.2500**	0.3750	0.5250	0.7000	0.9000	
11	0.2020	0.3030	0.4242	0.5656	0.7272	0.9090
12	0.1666	**0.2500**	0.3500	0.4666	0.6000	0.7500
13	0.1398	0.2097	0.2937	0.3916	0.5034	0.6293
14	0.1190	0.1785	**0.2500**	0.3333	0.4285	0.5357
15	0.1025	0.1538	0.2153	0.2871	0.3692	0.4615
16	0.0892	0.1339	0.1875	**0.2500**	0.3214	0.4017
17	0.0784	0.1176	0.1647	0.2196	0.2823	0.3529
18	0.0694	0.1041	0.1458	0.1944	**0.2500**	0.3125
19	0.0619	0.0928	0.1300	0.1733	0.2229	0.2786
20	0.0555	0.0833	0.1166	0.1555	0.2000	**0.2500**
21	0.0501	0.0751	0.1052	0.1403	0.1804	0.2255
22	0.0454	0.0681	0.0954	0.1272	0.1636	0.2045
23	0.0414	0.0621	0.0869	0.1159	0.1490	0.1863
24	0.0378	0.0568	0.0795	0.1060	0.1363	0.1704
25	0.0347	0.0521	0.0730	0.0973	0.1252	0.1565
26	0.0320	0.0480	0.0673	0.0897	0.1153	0.1442
27	0.0296	0.0444	0.0622	0.0829	0.1066	0.1333
28	0.0274	0.0412	0.0576	0.0769	0.0989	0.1236
29	0.0255	0.0383	0.0536	0.0715	0.0919	0.1149
30	0.0238	0.0357	0.0500	0.0666	0.8570	0.1071

TABLE II
PROBABILITY OF REVEALING THE LOCATION OF A STATIC USER FOR DIFFERENT VALUES OF n AND m

V. CONCLUSIONS AND FUTURE WORK

Location-Based Services will strongly emerge from the ICTs and they will become cornerstones of the new technology-based economies. In this paper, we have presented a concise overview of the state of the art of location privacy in Location-Based Services. In addition, we have recalled a novel TTP-free method and we have analysed its resiliency against a passive attack of several malicious users. It seems clear that using the chain-based approach improves the security of the system because all messages are not centralised by a

single user. In addition, the number of malicious users needed to obtain the location of a static user with high probability is large. So a reduced number of colluded users will not succeed if the number of non-malicious users is large enough. However, the chain-based approach has some problems that must be addressed in the future to let it be a success in real environments.

Some of these problems are the following:

- List integrity: The list of users involved in the distributed computation of the centroid can be modified by malicious users. It is necessary to define a method to avoid this attack or detect it.
- Contribution check: Malicious users can deliberately introduce noise in the chain to avoid the proper computation of the centroid. There is a need for a method to detect this behaviour.
- Protocol convergence: It must be guaranteed that the protocol finishes with a solution. Using time-outs seems to be a promising solution.

In addition, we plan to extend our statistical analysis to deal with different patterns of attack (e.g MSSM), and we are going to obtain simulation results.

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", TSI2007-65393-C02-02 "ITACA", CONSOLIDER CSD2007-00004 "ARES" and by the Government of Catalonia under grant 2005 SGR 00446.

References

[1] E. Union, "The Seventh Framework Programme (FP7)," 2006, http://ec.europa.eu/research/fp7/understanding/index.html.

[2] "Directive 2002/58/EC on privacy and electronic communications," http://europa.eu.int/eur-lex/pri/en/oj/dat/2002/201/20120020731en00370047.pdf.

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

[4] ——, "A ttp-free protocol for location privacy in location-based services," *Computer Communications (In press)*, 2008, (http://www.sciencedirect.com/science/article/B6TYP-4RNJ8NJ-1/2/7ed0fe3d12447f50d4344de4c8a2e3fa).

[5] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," SRI International, Tech. Rep., 1998.

[6] P. Samarati, "Protecting respondents' identities in microdata release," *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2001.

[7] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.

[8] M. Mokbel, "Towards Privacy-Aware Location-Based Database Servers," in *22nd International Conference on Data Engineering Workshops (ICDEW'06)*, 2006, pp. 93–103, atlanta, GA, USA.

[9] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: Architecture and algorithms," *IEEE Transactions on Mobile Computing (to appear)*, 2007.

[10] M. Gruteser and D. Grunwald, "Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking," *Proceedings of the First International Conference on Mobile Systems, Applications, and Services*, pp. 31–42, 2003.

[11] M. Mokbel and C.-Y. Chow, "Challenges in preserving location privacy in peer-to-peer environments," in *Seventh International Conference on Web-Age Information Management Workshops (WAIM '06)*, June 2006, pp. 1–8, hong Kong, China.

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

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

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

[15] H. Tschofenig, H. Schulzrinne, A. Newton, J. Peterson, and 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.

[16] "Geographic location/privacy (geopriv)," web, September 2006, http://www.ietf.org/html.charters/geopriv-charter.html.

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

[18] T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring," in *Advances in Cryptology EUROCRYPT'98*, ser. Lecture Notes in Computer Science, vol. 1403. Springer Berlin / Heidelberg, 1998, p. 308.