PROVIDING PRIVACY IN LOCATION BASED SERVICES[*]

Francesco Giudici, Elena Pagani, Gian Paolo Rossi
Information Science and Communication Department
Università degli Studi di Milano, Italy
Email: {fgiudici, pagani,rossi}@dico.unimi.it

ABSTRACT

In the last years, the availability of cheap mobile devices able to communicate via radio interfaces and equipped with low cost positioning technologies, like GPS, has brought to researcher attention a new kind of services: location based services (LBS). LBS require users to provide their position in order to customize the service.

This, of course, could be a threat for user privacy by at least two points of view. First of all, when a user performs a query to the LBS provider, she may disclose sensitive data. Imagine a user looking for a vegetarian restaurant: the LBS provider becomes aware that the issuer is vegetarian. Moreover, each query discloses user position, so the provider can track user location.

In this paper, we analyze existing approaches that address privacy issues in LBS, identifying their effectiveness on the basis of the knowledge contexts available to a potential attacker. A novel architectural framework is provided to analyze and classify proposed solutions on the basis of their connectivity and architectural requirements.

KEYWORDS

Wireless networks, Location Based Services, Privacy

1. INTRODUCTION

Privacy threat is due to LBS providers' untrustworthiness: a request sent to a provider should be somehow anonymized so that the service can be exploited by a user while her privacy is preserved.

The most obvious trick to put in practice is *ID anonymization*: the user identifier in the query is substituted by a pseudo-ID, so that the service provider can not bind sensitive data included in the request to the issuer. Unfortunately, this could be not enough to avoid user identification: an attacker can combine user position with some other external knowledge to identify the issuer. For example, consider a user issuing requests from her home. Also if she uses a pseudo-ID to hide her identity, with public available knowledge (yellow pages) an attacker can identify the issuer from her position. So, position should be somehow anonymized. The problem here is that user location should be disclosed to exploit a LBS. One solution could be *generalization*: a perturbed, less precise location is disclosed. Instead of providing exact user coordinates, a geographic area can be supplied.

1.1 ID anonymization

To provide privacy to users willingness using LBS we have first of all to substitute the identifier in the request sent to the LBS server. In literature, an anonymization server (AS) is generally supposed to be available: it will take in charge forwarding user requests to the LBS server (LBSS), replacing real identifiers with pseudo-ids. The LBSS then replies to the AS which forwards the reply back to the issuer. Several anonymization services are currently available on the Internet for free (proxy server). So, if LBS are provided over the Internet and users can access it, an AS can be easily found.

[*] This work has been partially funded by the Italian Ministry of University and Research in the framework of the "Context-Aware RouTing Over Opportunistic Networks (CARTOON)" PRIN Project.

1.2 Generalization algorithms

A function *f* that takes in input a set of coordinates <x,y> and gives back a generalized area A is called generalization function; an algorithm that computes a generalization function is called generalization algorithm. As suggested in [Mas], generalization algorithm goals are: (i) a high number of users should be indistinguishable from the issuer, (ii) the area A should be minimized to preserve service quality, (iii) computation of the algorithm should be efficient

Without extra knowledge, if K users are present in the area A, each one of them has an equal probability 1/K of being the issuer. This satisfy the definition of K-anonymity [CaSa]. The problem here is that the probability of each user being the issuer depends on the knowledge available to the attacker. So, generalization algorithms goodness analysis requires identification of the context, i.e. the knowledge, available to a potential attacker. [Mas], proposes a main division between static and dynamic knowledge contexts: in the dynamic case the attacker has the ability to make correlation between multiple requests of the same issuer. A similar division is then presented between the so called single-issuer and multiple-issuer cases: in the multiple-issuer case, a potential attacker is aware of requests from multiple issuers, and is able to gain information from the correlation of them.

In this work, we take into account just the static, single-issuer case. For this scenario, [Mas] models information available to the attacker in three main contexts:

1. C_{st}: attacker is aware of the geographic position of all users.
2. C_{st+g} : attacker is aware of the geographic position of all users and the generalization algorithm used.
3. C_{ast+g}: attacker is aware of the approximate user locations and the generalized algorithm used.

If a generalization algorithm guarantees K-anonymity in C_{st+g} it will be K-anonymous in C_{ast+g} too. So, we take into account only the first and the second cases.

2. ARCHITECTURAL CLASSIFICATION

In this work, we propose a classification of existing strategies addressing privacy issues in LBS, based on their connectivity and architectural requirements. Most solutions proposed in the literature assume a *centralized* architecture where a trusted anonymization server (AS) and a location aware trusted server (LTS) are available. The LTS is aware of the position of all users. When a user wants to issue a request to a LBSS, she establishes an encrypted connection with the AS and sends to it her issue, containing the real identifier, position and, of course, the query data (see Fig.1).

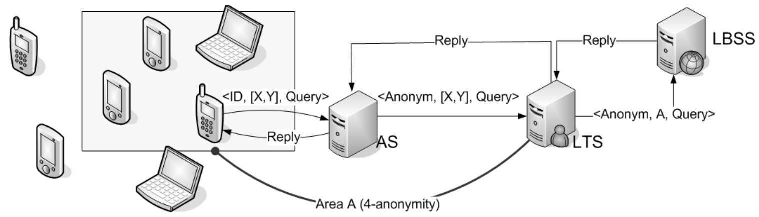

Fig. 1: Centralized Infrastructure. A user exploits an untrusted LBSS performing the query through an AS and a LTS.

The AS replaces the user identifier with a pseudo-ID and forwards the query to the LTS over an encrypted connection. The LTS will compute the generalization function on the position contained in the query and the user coordinates will be replaced by the estimated area A. The issue will be finally forwarded to the untrusted LBSS. The response will be forwarded back to the user through the LTS and the AS.

The two anonymization steps (ID substitution and generalization of the area) are generally performed by a single server (the LTS). So, from an architectural point of view, a single trusted server is required (LTS). Moreover, connectivity should always be available, allowing point-to-point connections between users and the LTS and between the LTS and the LBSS. We identify this architectural environment as A_{fc+a} (fully centralized trusted entity + always available connectivity). A_{fc+a} is a likely condition in a city environment: cellular service providers offer GPRS and UMTS always-on connectivity and have an efficient user tracking mechanism to be able to localize mobile callers requesting emergency assistance. So, they are the ideal candidates to provide the LTS service.

A *partially distributed* architectural model has been recently proposed in the literature: users cooperate to compute the generalization algorithm by themselves. However, a central trusted server is required for user authentication and to provide support to new joining users. Point-to-point connectivity is supposed to be available always and everywhere. We identify this environment as A_{pd+a}.

A more challenging environment is the one in which connectivity relies on multi-hop ad hoc communications among users: the mobile ad hoc networks environment (MANET) [RFC2501]. No GSM is supposed to be available, connectivity relies on nodes themselves, forwarding packets for each other. GPS can be used to acquire its own position. We will identify this environment as $A_{pd+manet}$.

3. SOLUTIONS PROPOSED IN LITERATURE

One of the first generalization algorithm ever proposed was designed to cope with C_{st} in an A_{fc+a} environment. It is named Interval Cloaking [IC]. The generalization algorithm is computed by the LTS itself which is in charge of managing all users in an area A. The LTS divides A in quadrants of equal size. Each quadrant q_{prev} is then split in quadrants *q* of equal size and so on. When the quadrant *q* where the issuer *u* is located contains less than K users, the quadrant q_{prev} is returned as the generalized position of *u*. Unfortunately, Interval Cloaking cannot guarantee K-anonymity in the C_{st+g} context, as showed in [PR].

A second generation of generalization algorithm has been proposed to overcome this problem. One for all, the dichotomicPoints algorithm [Mas] provides anonymity in the C_{st+g} context and A_{fc+a} environment. The algorithm iteratively partitions the whole managed area in two smaller areas, splitting the users in two groups of equal cardinality. The process is repeated until the resulting area contains less than 2K users. This will be returned as generalized area.

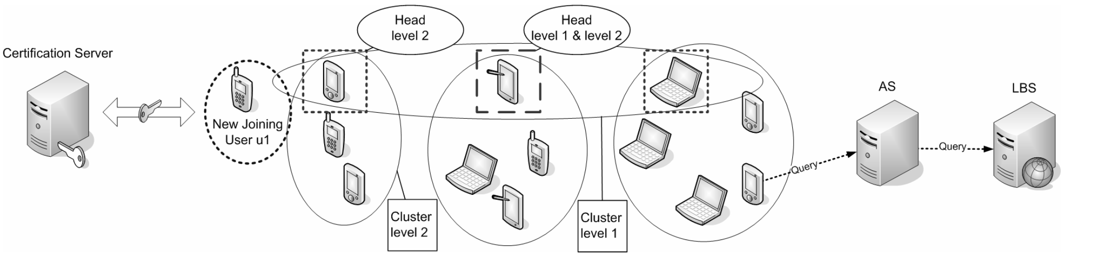

Fig. 2: Architecture of Privè. The computation of the generalization algorithm is performed in a distributed way.

Ghinita et al. have recently proposed Privè [PR], a distributed architecture that provides K-anonymity in a C_{st+g} context and a A_{pd+a} environment. In Privè, users organize themselves in clusters of level *i*. Each cluster_i elects a node-head among cluster members. Heads of clusters_i are then organized in a cluster_{i-1}, which in turn has a head_{i-1}. The process is repeated till a root head is established on the top of the whole hierarchy, as shown in Fig.2. The idea is to organize clusters as a B-tree, where nodes are sorted by their Hilbert value. This is computed through the HILBASR algorithm: using the Hilbert space-filling curve, user position is mapped to an integer. Then, users are grouped sequentially in buckets of K nodes: Moon et al. showed that with high probability two points close in the 2D space are close in the Hilbert transformation too [HilP], so buckets will include geographically close users. The generalized area for a issuer *u* is the MBR (Minimal Bounding Rectangle) enclosing the K nodes of the bucket to which *u* belongs. This satisfies the *K-ASR* (Anonymizing Spatial Region which encloses at least K users) Reciprocity definition:

a K-ASR A satisfies reciprocity iff exists a set of users AU lying in A such that:

$$(i) \quad |AU| \geq K \qquad\qquad (ii) \quad \forall u \in AU, \quad u \quad lies\ in \quad A$$

As shown in [PR], K-ASR reciprocity guarantees K-anonymity of the query source against location-based attacks (in the C_{st+g} context). So, K-ASR construction technique proposed in Privè guarantees K-anonymity.

In $A_{pd+manet}$ environment, each user is equipped with a low-medium range wireless radio device and should forward packets for other nodes. Users should be able to provide their position to the LBS provider: this means that a positioning device should be available to each node (e.g. GPS). In such environment, the most effective routing technique is Position Based Routing (PBR) [PBR]. In PBR forwarding decisions are made locally at each node, on the basis of the destination position and the coordinates of in radio range nodes. Destination position is provided through the *Location Service* (LS): it provides geographical coordinates of any possible destination in the network. Several LS strategies have been proposed in literature [LS]. Solutions that show best scalability are based on a hierarchical approach, based on the concept of *region*: the area to manage is split in smaller geographic areas (regions). A multi-level hierarchy is established, where each region has an head-node acting as head cluster, maintaining region membership of other nodes (a node is member of a region when is inside it). Head nodes are then connected to higher level nodes to construct a DNS-like architecture able to retrieve region membership of any node in the network [CAESAR]. In such scenario, privacy in the C_{st+g} context can be provided naturally: if at least K users are always present in a region, the region itself can be used as a K-anonymous generalized area. If a head-node realizes that nodes in its region are less than K, it can merge the region with a neighboring one to preserve K-anonymity, in a way similar to the resizing area operation presented in [CAESAR]. Anyway, C_{st+g} context is no more adequate for MANET environment. This problem has been recently investigated in literature, especially by researchers working on VANETs (Vehicular Ad hoc NETworks). Privacy regarding this environment has to cope with information disclosure due to PBR: it requires explicit disclosure of node position. Moreover, an attacker can bind data eavesdropped on the radio channel to geographically close nodes, analyzing the power of the radio signals received. Some proposals have been presented to cope with these hazards, e.g. the AMOEBA framework [AMB]. The task is very challenging, and they just enforce privacy without any guarantee on the effectiveness of the solutions proposed.

4. CONCLUSION

Many solutions have been proposed in the literature to deal with user privacy when exploiting Locations Based Services (LBS). We have seen that proposal effectiveness can be analyzed only if a framework classifying knowledge available to a potential attacker is present. Moreover, architectural solutions requirements should be taken into account too. Finally, we have had a quick look to a completely distributed environment, where communication relies on users themselves (MANET). While deployment of solutions to preserve privacy in presented formalized knowledge contexts has been successfully addressed, the adoption of this new paradigm of communication poses new threats due to the communication protocols and devices adopted: new knowledge contexts should be formalized in order to model attackers knowledge correctly. This is essential to allow planning and analysis of solutions that hopefully will be able to provide some guarantees (as K-anonymity) in totally distributed environments and ad hoc connectivity.

REFERENCES

[AMB] K. Sampigethaya, et al., 2007. AMOEBA: Robust Location Privacy Scheme for VANET. *Selected Areas in Communications, IEEE Journal on , vol.25, no.8, pp.1569-1589*

[CAESAR] F. Giudici, 2007. CAESAR: an urban location service for VANETs. *ACM SIGMOBILE Mobile Computing and Communications Review*

[HilP] B. Moon, et al., 2001. Analysis of the Clustering Properties of the Hilbert Space-Filling Curve. *IEEE TKDE.*

[IC] M. Gruteser and D. Grunwald, 2003. Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. *MobiSys*

[LS] T. Camp, et al., 2002. Location information services in mobile ad hoc networks. *IEEE International Conference on Communications, vol.5, no., pp. 3318-3324*

[Mas] S. Mascetti, 2007.Privacy Protection through Anonymity in Location-based Services. *DICo Technical Report n. 22-07, University of Milan*, Italy.

[PBR] I. Stojmenovic, 2002. Position-based routing in ad hoc networks. *IEEE Communication Magazine*.

[PR] G. Ghinita, et al., 2007. PRIVE: anonymous location-based queries in distributed mobile systems. *IWWW.*

[RFC2501] S. Corson, J. Macker, 1999. RFC2501: Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. Work in Progress.

[SaSw] S. De Capitani di Vimercati, P. Samarati, 2006. k-Anonymity for Protecting Privacy. *in Information Security*.