Private Proximity Testing For Location Based Services

L. Ertaul, A Balluru, A. Perumalsamy

Math and Computer Science Department, California State University, East Bay
California, USA
Levent.ertaul@csueastbay.edu, aballuru@horizoncsueastbay.edu, aperumalsamy@horizoncsueastbay.edu

***Abstract*— Over time privacy attacks on the Internet and Internet–attached systems have grown sophisticated and attacks have become more automated and can cause greater amounts of damage. Thus, a wide range of technologies and tools, complex protocols and applications are needed to counter the growing threat. This paper deals with the implementation and analysis of private proximity testing in the context of Location Based Services (LBS). The protocol states that Alice and Bob can investigate their proximity by exchanging set of encrypted messages via the server. The approach is novel since the server will not be able to track either Alice or Bob.**

***Index Terms*—Location Based Services, Proximity Testing, Privacy.**

I. INTRODUCTION

Location Based Services (LBS) are ubiquitous in today's applications. They are an indispensable component of our communication model as LBS has proven to be crucial not only for the companies but also for the consumers. Tracking and monitoring individuals, children and thieves and its uses in the law enforcement by police really has good implication on the society. In case of business as a vehicle tracking device or asset tracking component, LBS technology acts as a catalysts in the growth of industries especially telecommunication and transportation. However, as the system deals with confidential personal information like location, personal mobile number, concerning address, it becomes vital for the operator to offer adequate security for maintaining user's privacy [3, 4, and 10].

LBS, besides providing numerous services to consumers worldwide, they are also notorious in collecting user activity. This helps them target specific products at individuals which is a market proven strategy for increased growth. Vendors of many of the mobile applications often exploit the data that is collected by the use of their services. Location-based service advertising -- which ties in consumer locations with restaurants, retail shops and other locations through mobile devices -- will grow to over one-third of all mobile advertising in four years [14, 15 and 16].

According to a study by Pyramid Research [17], location-based revenue in the US is expected to climb from $2.8 billion in 2010 to $10.3 billion in 2015. By 2015, location-based advertising will be $6.2 billion, according to Pyramid Research. In 2010, location-based advertising was $588 million -- 18.5% of all mobile advertising. Location-based advertising will generate 60% of all location-based revenue in four years. Pyramid analyst Jan ten Sythoff believes that all forms of mobile advertising will grow. "However, local search will be the most important driver of location-based advertising revenues." Not only the developers of navigation applications will be changing their business model to fit into the local-search branch, but many different companies from different branches can also profit from the growth the of the local-search market – from start-ups like Poynt and Yelp, to the local business advertising from specialized portals like the Yellow Pages, to even the search engines that are specialized for a particular topic, like toptable or HotelBooker. The survey conducted by Pyramid shows the amount of revenue generated by the use of LBS.

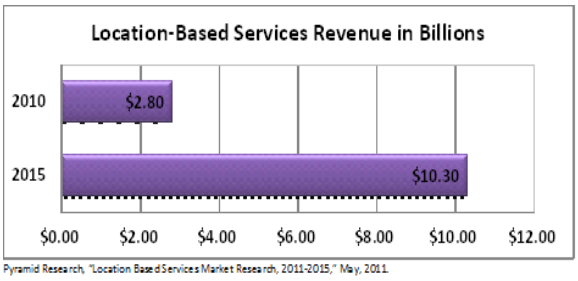

Fig 1.1LBS Services Revenue

The figure contrasts the revenue generated in 2010 vs. the projected estimate for the year 2015. It is observed that this amount is indeed staggering. This definitely suggests that data mining resulting from the use of LBS is a big boost to their revenues.

When is a person permitted to monitor someone by using LBS? Should the concerned person's consent is necessary? What about individual right and personal autonomy? What kinds of evidences are required to monitor a person? These are some of the questions that need to be answered before using LBS as monitoring a person can have psychological effect on the person being monitored. In case of monitoring criminals or suspects by police or security agencies the question of individual freedom came, as enforcing someone's freedom is not at all ethical when the person is only suspected of committing the crime. We notice a varying level of concern between male and female users. This is shown in Fig 1.2

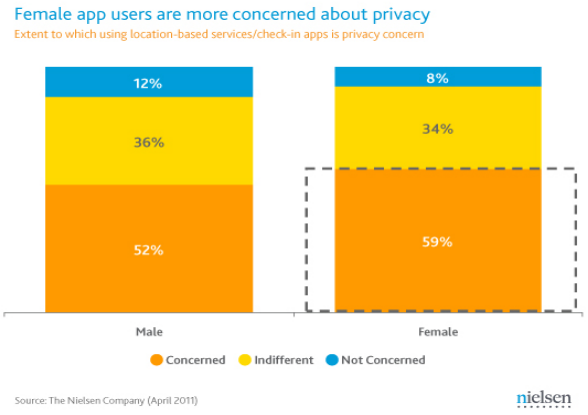

Fig 1.2 Level of privacy concern between male and female users

However, in the whole privacy and security issues of LBS, there are chiefly four points came as control, trust, privacy and security as legal, social, ethical and technological aspects. But all four are mutually exclusive as control decreases trust, trust enhances privacy, which needs security, and security again increases control.

Control (Legal) – Commonly GPS and other LBS devices are used to control and offer various types of services to the user. Personally it controls one's own direction of moving in guiding the right way. In case of child tracking, parents have exclusive right to look after their children, as it is not possible for the young ones to make their own decision. So it is their legal right to monitor their children thereby reflecting a sense of caring. In case of law enforcement, special laws provide legal rights to police or security departments to keep an eye on criminals or suspect.

Trust (Social) – In social life trust is the most essential part in human relationship. However, the use of LBS is being practiced in low trust conditions. Monitoring someone with the help of tracking system really affects personal relationship but as far as tracking criminals by cops or tracking children by parents are concerned, it is for the welfare of the individual & society.

Privacy (Ethical) – As a human being, everyone has the right to privacy or being free from intrusion or disturbances in one's personal life. But in case of LBS or any other telecommunication technologies dealing with transformation of various kinds of information, it becomes essential to provide adequate security to these kinds of data for not being misused by any unauthorized person. Tracking and monitoring someone without his/her consent is purely unethical so needs high level of security. But again as in case of law and order where tracking devices are used to monitor criminals becomes essential for the sake of society as a whole. Here, social security is counted higher than Individual safety and security.

Security (Technological) – Again for maintaining privacy, security system should be strong. Every technology has both positive and negative impact on human life and LBS also has shortcomings by locating accurate information data or even easily given access to unauthorized person. On one hand LBS enhances both national and personal security but create another problem for the privacy of individual by not providing a foolproof security system to that highly sensitive information stored in its database. For obtaining security, one needs to do a little compromise on his/her privacy but to what extent is a question. Fig 1.3 below shows is a survey conducted to gauge the concerns of various smart phone users and their use of LBS Despite the various concerns as posed by LBS, it is still considered as an invaluable tool for efficient communication. Following section will contrast the risks and benefits of LBS.

II. FAST ASYNCHRONOUS PRIVATE PROXIMITY TEST WITH AN OBLIVIOUS SERVER

There are a variety of Private Equality Testing protocols available [13, 14, and 15]. This protocol is a novel approach proposed and implemented by a team of researchers at the Stanford University. This protocol unlike its variant operates asynchronously, which implies that the two parties do not have to be online at the same time. In other words, it can be seen as a two party protocol at any instant of time where the interaction happens between a client and the server. In this setting, the server is responsible for not only handling the transactional requests but also performing some mathematical operations that will be discussed shortly.

The reason for making server more involved in this form of testing is to prevent dictionary attacks. This attack is possible if the application was a mere non-interactive message transferring between two parties exchanging only hash of their

locations. The asynchronous setting allows a privacy-efficiency tradeoff due to the fact that the sender and receiver can each execute their half of the protocol with the server at arbitrary times. Specifically, a user might configure her/his device to participate in the protocol in the role of sender only when her location changes. Similarly, she/he might configure her device to participate in the role of receiver only when she explicitly checks the proximity testing application. For the sake of the explanation, Bob would be the sender side of the client application and Alice would be the receiver side of the application.

The protocol requires Alice and Bob to generate quantized location values which is representative of the center of the grid that they belong to. The protocol states that if their quantized locations match, Alice can know that they are in the same grid otherwise she can only know that Bob is in a different location than her location and nothing else. Bob does not learn anything in this process. It is assumed that the keys K_{ab}, K_b, K_a shared between Alice and Bob, Bob and the Server, Alice and server respectively are distributed using the concept of Social Keys[12].

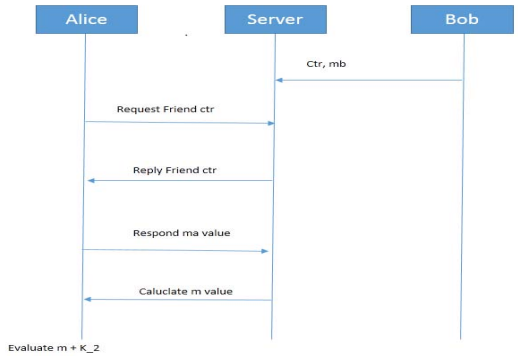

Fig 1.3 Private Proximity Testing

Application Set Up: The protocol generates 64 bits of quantized location data which is a function of the actual location of the client. It then generates three 64 bit parameters ***K_1, K_2*** and ***r*** using a secure pseudo-random function ***F (k, x)***, where ***K*** is the key for the secure pseudo-random function and ***x*** is the counter value that will be used to generate the pseudo-random number. Our implementation uses AES in ECB mode as the pseudo-random function [11]. There are three categories of messages that are exchanged in this protocol as shown in figure 1.3

Message 1: Bob computes the counter value as a function of time of day which is a 64 bit data. The counter (***ctr***) is incremented by one every time the pseudo-random function is called to generate ***K_1, K_2*** and ***r*** values. This is shown as follows:

$$K\_1 = F(K_{ab}, ctr + 1);\ K\_2 = F(K_{ab}, ctr + 1)$$

$$R = F(K_b, ctr + 1)$$

Bob then masks his quantized location ***Bt*** as ***Mb = r (Bt + K_1) + K_2*** and transmits this message along with the counter value. Server on receiving the counter value calculates r and stores it.

Message 2: This step is initiated by Alice when she desires to query about Bob. She first sends a query message to the server asking for Bob's counter value. The server looks up the request in its database and sends the counter value of the friend if it is fresh. On receiving the response from server, Alice then checks if the counter is fresh or not. If it is fresh, she calculates ***K_1, K-2*** using the pseudo-random function with the key she shares with Bob. Essentially, the ***K_1*** and ***K_2*** generated by Bob must match her values. She then masks her quantized location ***At*** by calculating ***Ma = At + K_1***. She then sends this value to server.
Message 3: Server on receiving the message from Alice calculates a value m as:

$$m = r*ma - m2$$

Alice receives this response and computes ***m + K_2***. If this yields a value of ***0***, it implies that Alice and Bob are close by or rather in the same grid. The value comes to be zero only if the quantized locations match. This is illustrated as below:

$$\begin{aligned} m &= r*ma - mb \\ &= r*(At + k1) - r*(Bt + k1) - k2 \\ &= r*(At - Bt) + k2 \end{aligned}$$

A non-zero value only implies that they are not close to each other and nothing else. Bob is not intimated about anything. He is not aware if anybody is looking for him. The following section will deal in detail regarding the various design decisions taken and the challenges faced. The performance and security of our implementation is also measured.

III. IMPLEMENTATION MODEL

Platform Details – We developed the protocol as a TCP/IP client server model on Java platform on Windows 7 operating system. Java's crypto library provided the AES based pseudo-random function. The key sizes were 128 bits. We have used JAVA SWT to implement the GUI for the application. The table below summarizes the platform details of our implementation.

Table 1.1 Implementation Platform Details

Runtime Environment	Java SE 1.6
Network Model	TCP/IP Client Server Model
Platform	NetBeans
GUI	Java SWT
Programming Language	Java
Operating Systems	Windows 7
Crypto Library	Javax.Crypto
Pseudo Random Function	AES – 128 bits ECB

Key sizes	128 bits
K_1,K_2,r,mb,ma,ctr sizes	64 bits

Message Formats – The client application communicates with the TCP client via messages in IP format. Each packet is of 25 bytes in size. For various messages following table 1.2 describes the transmission message format.

Table 1.2 Transmission Message Formats

Message Number	Packet Type	Description
Message A	3	The client queries for its friend's location data. If the requested friend's counter value is available, the server responds the same to Alice.
Message B	5	The client performs calculations and sends the same to the server. Expects a response with value m from the server.
Message C	7	The client wishes to update the location to the server and sends its counter and mb value.

The first byte of the messages sent from the client to the server denotes the packet type. All the messages contain the user ID of the initiator. The generic format of the message from the client to the server is as follows:

Table 1.3 Format of Client-Server Message

Packet Type	Client ID	Friend ID	Payload

All packets from the client have packet IDs of 3, 5, and 7. If a packet of any other ID is received, it is discarded.

Ideal Conditions vs. Assumptions– We have attempted to model various real life scenarios to a large extent; however, there is a scope of incorporating many more features to make the application more versatile. This section deals with the design decisions taken for the sake of the implementation and a possible solution that would be more suitable for real life application.

1. **Key Distribution** – Our application requires three secure 128 bit keys to be distributed K_{ab}, K_b, K_a which is shared between Alice and Bob, Server and Bob, and Server and Alice respectively. The secure distribution of these keys is vital for the security of this protocol since they are used to in the secure pseudo-random generator (AES) [11]. The level of security in this field will ensure to keep dictionary attacks at bay. In our implementation we have assumed that the keys have already been distributed securely and that all the parties know the keys required by each other. One of the ideal methods of key distribution will be the concept of Social Keys. SocialKeys embraces the idea that public keys must be associated with the digital identities that people widely own use, i.e., their social network accounts, rather than requiring the creation of new identities for cryptographic purposes. Although not novel in its approach, this may seem to be a more viable option as there is no involvement of a third party for maintenance of the keys. Instead various features of the Social Network are repurposed in order to achieve this. By offloading the establishment of trust between users to the social network, SocialKeys obviates the need for a "key manager". As a consequence, it is almost completely transparent to the user.

2. **User IDs** – On the same lines as the key distribution, the user IDs have been manually assigned to the users. It is given as an input when the application is fired up. Ideally, the user IDs would pertain to the social network user account since the key retrieval also happens from SocialKeys.

3. **Quantized Location Data –** The application requires generating quantized location data. This data is obtained by dividing an area of certain range into overlapping hexagonal grids as shown in the Fig 1.5. The grids are represented by the center of the grid. Users belonging to any one grid will have a quantized location that is referred to the center of the grid that they belong to. If Alice is in grid marked x and her friends in the same grid, then the protocol results in letting Alice know that they are in the same grid or not. Other users are not aware of anything. The use of quantized location further masks the actual location of the user and is way of representation of the location. This quantized location is represented by 64 bits of data [20, 21]. The grids themselves, ideally must be allowed for user configuration [5]. Fig 1.4 is as below.

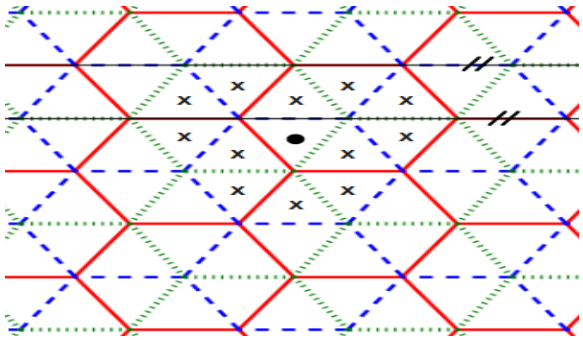

Fig 1.4 Grid structure for user configuration

However, in our implementation, we are randomly generating the quantized locations and feeding it to the application.

4. **Boundary Conditions –** The proximity testing allows applications to test for proximity besides equality. This allows users to determine the relative closeness even if they don't belong to the same grid. This is especially useful as it test for the boundary conditions. This is discussed at length in [1]. Based on the grid structure shown in Fig 1.4.However, in our application we would be testing for equality where in if the users are in the same grid, they are said to be close.

Performance Analysis – The protocol was executed on Windows environment with Intel core i3 with a processor speed 4 GHz. A very comprehensive study of the performance of the protocol was conducted. As noted by the authors of the protocols and in my research, I have concluded that the protocol is definitely an improvement over the synchronous version of the protocol. There are many reasons that contribute towards this agenda.

A. Synchronous vs. Asynchronous mode of execution – Unlike synchronous counterpart of this protocol, the clients after having updated their locations can go offline and be still probed for proximity by their friends while being offline. This reduces tremendous overhead on the communication network as there is no necessity of clients being connected to the server at all times.

B. 64 bits of data for communication – This implementation requires 64 bits of data for representing data values of counter, AES generated random values, quantized location data, masked location data. A 64 bits of data representation was used as it is established for security purposes. Also the effective bytes per edge is still 8 bytes compared to 4 bytes if 32 bits of data was used. This is not as much of an overhead still.

C. **Delay Analysis –** The delay analysis is a crucial aspect of the protocol as it justifies the design decisions. The following table describes the delay analysis on the client side that updates its location data to server.

Table 1.4

Attempt #	Delay (milliSec)
1	358
2	258
3	263
4	308
5	297

The following table describes the delay analysis on the client side that queries the counter value of the friend from the server

Table 1.5

Attempt #	Delay (milliSec)
1	255
2	269
3	247
4	303
5	266

The above results are reflective for a setup of 2 client and one server system. However, based on the result, we can extrapolate the result that it may take about 2.5 seconds to 3 seconds on a computer with this environment to execute for about 100 friends. This can be established since the messages get multiplexed and sent as a package to the server and the same is received from the server. Hence there will be no more overhead in establishing connection with 100friends than it is for 1 friend.

Security – The security of a system is always under attack when there is an involvement of a party besides the participants of the application. In our case, as has already been stated, the server is involved in processing of some of the messages and computing values that help Alice determine proximity. Even though the quantized locations are masked, there are two possibilities of attacks that can be immediately seen.

First is the case when the server and Bob collude. If so, Bob can easily estimate Alice's location. Likewise, if Alice colludes with server, Bob's location can be estimated.

Challenges – There were many challenges that were encountered in the implementation of the protocol**.** Following is a list of few of them:

a. Configuring of the TCP/IP server
b. Determining the packet format for transaction.
c. Determining number of bytes for each field.
d. Assigning keys to various users.
e. Integration with GUI
f. Determining the mode of operation of AES function.

Screenshots: Following are some screenshots for the execution of the protocol. Figure below shows the scenario when Bob updates his location with server with an ID of 1000.

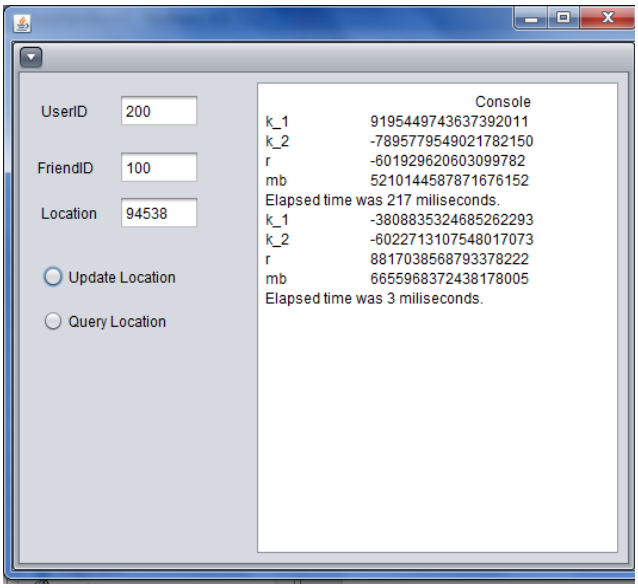

Fig 1.5 Result window for Location Update

Following is a figure that shows the execution of querying of location of Bob by Alice.

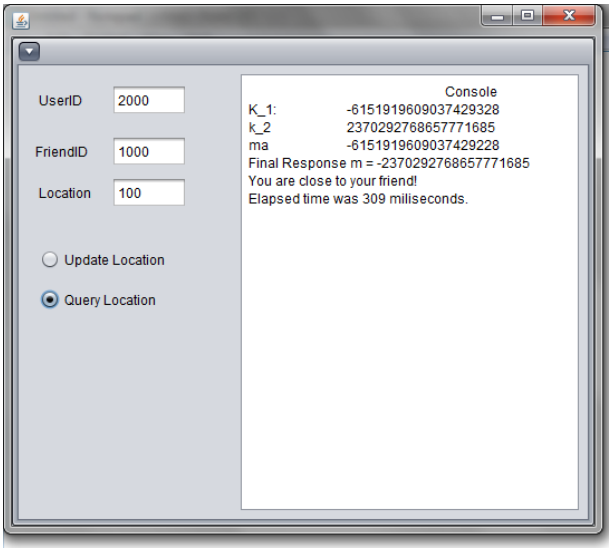

Fig 1.6 Result window for Proximity Testing

IV. Conclusions

Location-based social networks carry user-driven geographical information, and bridge the gap between real-world and online social media. In this paper we implemented Asynchronous proximity testing protocol without revealing actual location information of the user. This protocol does not leak any information about the secret value thus and preserves privacy of the user however there are still some issues to be resolved.

Instead using random number representation for the location data, a real time location data can be obtained from the LBS like Loopt, Google Latitude etc. Moreover session events can be developed to preserve previous activities and user can be requested to have login and password information because, If no session events are stored, So If server or client goes down will cause loss of data and new session should be started to run the system.

It is very important to ensure the privacy of the data. There has been extensive research in the field of providing security to many aspects of LBS services [19], [20].

References

[1] Narayanan, Thiagarajan, Lakshmi, Boneh, Hamburg, on Location Privacy via proximity testing, IEEE

[2] S. Jarecki, X.Liu, on Efficient Oblivious Pseudorandom Function IEEE, 2009

[3] R. D. Hopkins, R. Ho, I.E. Sutherland on Proximity Communications, IEEE, 2004.

[4] F.Olumofin,Tysowoski,Goldberg,Hengartner on Achieving Efficient Query Privacy for Location Based Services, IEEE, 2010

[5] C.Gentry, Fully Homomorphic encryption using ideal lattices, IEEE,2009

[6] M. Naor, B. Pinkas on Oblivious transfer and polynomial evaluation, IEEE, 1999.

[7] M. Raya, J.P Hubaux on The Security of Ad Hoc Networks, IEEE, 2005

[8] Rahman, S. Halles on A Distributed tust Model, New Security Paradigms, 2006

[9] Drost,R. Forrest, C. ; Guenin, B. ; Ho, R. ; Krishnamoorthy, A.V. ; Cohen, D. ; Cunningham, J.E. ; Tourancheau, B. ; Zingher, A.; Chow, A. ; Lauterbach, G. ; Sutherland, I on High Performance Interconnects, IEEE, 2005.

[10] Kuper A; Treu G; on Efficient proximity and separation detection among mobile targets for supporting location-based community services; ACM Digital Library, 2006

[11] J. Daemen and V. Rijmen, AES Proposal: Rijndael, AES Algorithm Submission, published and accepted in http://csrc.nist.gov/publications/fips/

[12] R. McGeehan, My Public Key (Facebook application) http://www.facebook.com/apps/application.php?id=7923770364

[13] F. Boudot, B. Schoenmakers, and J. Traore. A fair and efficient solution to the socialist millionaires' problem.Discrete Applied Mathematics, 2001

[14] R. Fagin, M. Naor, and P. Winkler. Comparing information without leaking it. Comm. of the ACM, 39:77– 85, 1996.

[15] S. Mascetti, D. Freni, C. Bettini, X. S. Wang, and S. Jajodia, “Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies,” VLDB J.

[16] L. Siksnys, J. R. Thomsen, S. Saltenis, and M. L. Yiu, “Private and flexible proximity detection in mobile social networks,” in Mobile Data Management, T. Hara, C. S. Jensen, V. Kumar, S. Madria, and D. Zeinalipour-Yazti, Eds. IEEE Computer Society, 2010.

[17] http://www.pyramidresearch.com/store/Report-Location-Based-Services.htm

[18] Data Analysis on Location-Based Social Networks, Huiji Gao and Huan Liu white paper.

[19] Analysis of a Location-Based Social Network, Chen Guanling, Computational Science and Engineering, 2009. CSE '09. International Conference, Aug. 2009

[20] M. Freedman, K. Nissim, and B. Pinkas. Efficient private matching and set intersection. In Proc. of Eurocrypt’ 04, pages 1–19. Springer-Verlag, 2004.

[21] C. Hazay and Y. Lindell. Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries. In Theory of cryptography (TCC), pages 155–175, 2008.