ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

Secure Privacy Assistance Content Protecting Location Based Queries Using Symmetric Key

R.Abijah Dionysius [1], D.Vignesh Kumar [2], P.Venkatesh [3], Dr.K.Saravanan,B.E.,M.Tech.,Ph.D [4]

B.E Final year Student, Dept. of Computer Science and Engineering, Pavai College of Technology, Tamilnadu, India [1]

B.E Final year Student, Dept. of Computer Science and Engineering, Pavai College of Technology, Tamilnadu, India [2]

B.E Final year Student, Dept. of Computer Science and Engineering, Pavai College of Technology, Tamilnadu, India [3]

Associate Professor & Head of the Dept, Dept. of CSE, Pavai College of Technology, Tamilnadu, India [4]

ABSTRACT: Location based system are used for finding out point of interests (POI) from a specific location. Usually a GPS latitude and longitude is sent as an input to the location servers and based on the GPS coordinate the point of interests can be served back to the client from the location server. In the project we proposed to solve problems associated with the location data. The user does not want to send his location data (GPS coordinate) to the server directly, since doing so the server can find the user's location preferences and use that data for advertising the user's privacy is lost. The second part is like the server wants to protect its data from the user query. The server want to return back only relevant data to the user .The server cannot sent back other sensitive data to the user. We propose a major enhancement upon previous solutions by introducing a two stage approach, where the first step is based on Enhanced Symmetric key Transfer and the second step is based on Private Information based on Enhanced Symmetric key Retrieval, to achieve a secure solution for both parties. The solution we present is efficient and practical in many scenarios. We implement the solution using a real cloud location server and android mobile application. . (1) The system only requires a semi-trusted third party, responsible for carrying out simple matching operations correctly. This semi-trusted third party does not have any information about a user's location. (2) Secure snapshot and continuous location privacy is guaranteed under our defined adversary models. (3) The communication cost for the user does not depend on the user's desired privacy level; it only depends on the number of relevant points of interest in the vicinity of the user. (4) Although we only focus on range and k-nearest-neighbor queries in this work, our system can be easily extended to support other spatial queries without changing the algorithms run by the semi-trusted third party and the database server, provided the required search area of a spatial query can be abstracted into spatial regions. Experimental results show that our DGS is more efficient than the state-of-the-art privacy-preserving technique for continuous LBS.

KEYWORD: point of interests (POI), GPS, LBS.

I. INTRODUCTION

Methods have also been proposed to confuse and distort the location data, which include path and position confusion. Path confusion was presented by Hoh and Gruteser. The basic idea is to add uncertainty to the location data of the users at the points the paths of the users cross, making it hard to trace users based on raw location data that was k-anonymised. Position confusion has also been proposed as an approach to provide privacy. The idea is for the trusted anonymiser to group the users according to a cloaking region (CR), thus making it harder for the LS to identify an individual. A common problem with general CR techniques is that there may exist some semantic information about the geography of a location that gives away the user's location. For example, it would not make sense for a user to be on the water without some kind of boat. Also, different people may find certain places sensitive. Damiani *et al.* have presented a framework that consists of a obfuscation engine that takes a users profile, which contains places that the user deems sensitive, and outputs obfuscated locations based on aggregating algorithms. As solutions based on the use of a central anonymiser are not practical, Hashem and Kulik presented a scheme whereby a group of trusted users construct an ad-hoc network and the task of querying the LS is delegated to a single user . This idea improves on the previous work by the fact that there is no single point of failure. If a user that is querying the LS suddenly goes offline, then another candidate can be easily found. However, generating a trusted adhoc network in a real world scenario is not

DOI: 10.15680/IJIRCCE.2016. 0403155

always possible. Another method for avoiding the use of a trusted anonymiser is to use ‘dummy’ locations. The basic idea is to confuse the location of the user by sending many random other locations to the server, such that the server cannot distinguish the actual location from the fake locations. This incurs both processing and communication overhead for the user device. The user has to randomly choose a set of fake locations as well as transmitting them over a network, wasting bandwidth. We refer the interested reader to Krum, for a more detailed survey in this area.

II. RELATED WORKS

This paper tackles a major privacy threat in current location-based services where users have to report their exact locations to the database server in order to obtain their desired services. For example, a mobile user asking about her nearest restaurant has to report her exact location. With untrusted service providers, reporting private location information may lead to several privacy threats. In this paper, we present a peer-to-peer (P2P) spatial cloaking algorithm in which mobile and stationary users can entertain location-based services without revealing their exact location information. The main idea is that before requesting any location-based service, the mobile user will form a group from her peers via single-hop communication and/or multihop routing. Then, the spatial cloaked area is computed as the region that covers the entire group of peers. Two modes of operations are supported within the proposed P2P spatial cloaking algorithm, namely, the on-demand mode and the proactive mode. Experimental results show that the P2P spatial cloaking algorithm operated in the on-demand mode has lower communication cost and better quality of services than the proactive mode, but the on-demand incurs longer response time.

III. EXISTING SYSTEM

The existing system uses two stage approaches to preserve privacy for users and location service provider and implements better solution for privacy over two communications like users to service provider and service provider to location server.

In Existing system a major enhancement upon previous solutions by introducing a two stage approach, where the first step is based on Oblivious Transfer and the second step is based on Private Information Retrieval, to achieve a secure solution for both parties. The solution we present is efficient and practical in many scenarios. We implement the solution using a Google place location server and android mobile application.

Disadvantages

- LS (Location Server) supplying misleading data to the client.
- This misleads about integration of all the model
- Less security for mobile users
- To affect them privacy for mobile users

IV.PROPOSED SYSTEM

The proposed system uses a real cloud implementation of the location server. The data which server has is completed protected from the user. As well the location information (GPS coordinate) of the user is never sent directly instead only grid information will be sent. We will be implementing the client using an android mobile.

ADVANTAGES

- We overcome the data misleading between location server and users.
- We provide better security algorithm to protect user’s information during transformation.
- Real-time using cloud and android mobile.

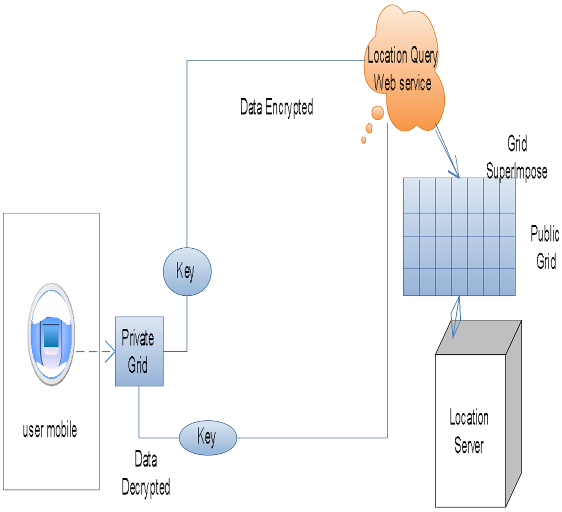

Fig 1: Architecture Design

IV. ALGORITHM

Advanced Symmetric key Algorithm

Advanced Symmetric-key algorithms are a class of algorithms for cryptography that use trivially related, often identical, cryptographic keys for both decryption and encryption etc.

The encryption key is trivially related to the decryption key, in that they may be identical or there is a simple transformation to go between the two keys. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link.

Other terms for symmetric-key encryption are **secret-key**, **single-key**, **shared-key**,**one-key**, and **private-key** encryption. Use of the last and first terms can create ambiguity with similar terminology used in public-key cryptography.

Contents

Types of symmetric-key algorithms

Symmetric-key algorithms can be divided into stream ciphers and block ciphers. Stream ciphers encrypt the bits of the message one at a time, and block ciphers take a number of bits and encrypt them as a single unit. Blocks of 64 bits have been commonly used. The Advanced Encryption Standard (AES) algorithm approved byNIST in December 2001 uses 128-bit blocks.

Some examples of popular and well-respected symmetric algorithms include Twofish, Serpent, AES (Rijndael), Blowfish, CAST5, RC4, 3DES, and IDEA.

Cryptographic primitives based on symmetric ciphers

Symmetric ciphers are often used to achieve other cryptographic primitives than just encryption.

Encrypting a message does not guarantee that this message is not changed while encrypted. Hence often a message authentication code is added to a cipher text to ensure that changes to the cipher text will be noted by the receiver. Message authentication codes can be constructed from symmetric ciphers (e.g. CBC-MAC).

However, symmetric ciphers also can be used for non-repudiation purposes by ISO 13888-2 standard.Another application is to build hash functions from block ciphers. See one-way compression function for descriptions of several such methods.

VI. METHODOLOGY

Getting User grid cell information

- Finding Server information
- Point of interest
- Symmetric encryption
- Sending information without gps

Getting User grid cell information

Android Mobile application will send the user grid cell information to the server and the security key to decrypt the data. We will be using symmetric encryption key. *Function*: user will search for information asking for the Point of interest say a shopping mall nearby or any route from source to destination The location provider will return the point of interest to the user queries by displaying nearby attraction for given query. The server will return the data based on search with its gps coordinates. The client will send a request using the server's cell information asking for the Point of interest say a shopping mall nearby. With the knowledge about which cells are contained in the private grid, and the knowledge of the key that encrypts the data in the cell, the user can initiate a private information retrieval protocol with the location server to acquire the encrypted POI data. Assuming the server has initialized the integer *e*, the user *ui* and *LS* can engage in the following private information retrieval protocol using the *ID*, obtained from the execution of the previous protocol, as input. The *ID* allows the user to choose the associated prime number power πi, which in turn allows the user to query the server.

2. Finding Server information

The server will super imposed the user cell information with the server location grid and find the location server's grid cell information. The server's cell Information will be sent back to user.

The server will also send all the information encrypted using the key sent by the client. The communication between client and server will be secure. The client will never send the GPS coordinate but will send an information grid of its location. New privacy metrics have been proposed that captures the users' privacy with respect to LBSs. The authors begin by analyzing the shortcomings of simple k-anonymity in the context of location queries. Next, they propose privacy metrics that enables the users to specify values that better match their query privacy requirements. From these privacy metrics they also propose spatial generalization algorithms that coincide with the user's privacy requirements. Methods have also been proposed to confuse and distort the location data, which include path and position confusion. Path confusion was presented by Hoh and Gruteser. The basic idea is to add uncertainty to the location data of the users at the points the paths of the users cross, making it hard to trace users based on raw location data that was k-anonymised. Position confusion has also been proposed as an approach to provide privacy. The idea is for the trusted anonymiser to group the users according to a cloaking region (CR), thus making it harder for the LS to identify an individual. A common problem with general CR techniques is that there may exist some semantic information about the geography of a location that gives away the user's location. For example, it would not make sense for a user to be on the water without some kind of boat. Also, different people may find certain places sensitive. Damiani *et al.* have presented a framework that consists of a obfuscation engine that takes a users profile, which contains places that the user deems sensitive, and outputs
Obfuscated locations based on aggregating algorithms

3. Point of interest

The client will send a request using the server's cell information asking for the Point of interest say a shopping mall nearby. The server will return an encrypted data of the point of interest to the client. The client will plot the information in the graph. *Function:* location of the data will be identified in private grid which is generated internally by the application there by client will sent his/him location to be determined in public grid which is on the server side The client will send a request using the server's cell information asking for the Point of interest say a shopping mall nearby, then the server will return an encrypted data of the point of interest to the client.

4. Symmetric encryption

The request sent by the client will use the symmetric encryption key to encrypt all the data. Further the connection between server and client will be secure. The request sent by the client will use the symmetric encryption key to encrypt all the data. Further the connection between server and client will be secure. the privacy of the user is maintained by constantly changing the user's name or pseudonym within some mix-zone. It can be shown that, due to the nature of the data being exchanged between the user and the server, the frequent changing of the user's name provides little protection for the user's privacy. A more recent investigation of the mix-zone approach has been applied to road networks. They investigated the required number of users to satisfy the unlink ability property when there are repeated queries over an interval. This requires careful control of how many users are contained within the mix-zone, which is difficult to achieve in practice. A complementary technique to the mix-zone approach is based on k-anonymity. The concept of k-anonymity was introduced as a method for preserving privacy when releasing sensitive records. This is achieved by generalization and suppression algorithms to ensure that a record could not be distinguished from $(k-1)$ other records. The solutions for LBS use a trusted anonymiser to provide anonymity for the location data, such that the location data of a user cannot be distinguished from $(k-1)$ other users.

Encryption

```
1. Given Plain Text.
 2. Randomly generate key k
 3. Calculate key K2 and key 3 from the key k.
 4. Repeat
5. Divide the n bits of plain text P into r multiple blocks of key size k such that n = k * r + m where m is mod (n,k)
6. Shuffle r blocks using key k
 7. Substitute the text (n bits) using key k2
 8. Shift the text in circular left shift with k3
9. until all round done
```

Decryption

```
1 Given Cipher Text and key k
 2. Calculate key k_inv , k2 & k
3 from the key k. 3. Repeat
 4. Shift the text in circular right shift with k3
5. Substitute the text (n bits) using key k2
6. Divide the n bits of plain text P into r multiple blocks of key size k such that n = k * r + m where m is mod(n,k)
 7. Shuffle r blocks using inverse key k_inv
 8. until all round done.
```

5. Sending information without gps

The server will also send all the information encrypted using the key sent by the client. The communication between client and server will be secure. The Client will never send the GPS coordinate but will send an information grid of its location. We analyze the security of the client and the server. While the client does not want to give up the privacy of his/her location, the server does not want to disclose other records to the client. This would not make much business sense in a variety of applications. the information that is most valuable to the user is his/her location. This location is mapped to a cell. In both phases of our protocol, the oblivious transfer based protocol and the private information retrieval based protocol, the server must not be able to distinguish two queries of the client from each other. We will now describe both cases separately.

VII. CONCLUSION AND FUTURE WORK

We have presented a location based query solution that employs two protocols that enables a user to privately determine and acquire location data. The first step is for a user to privately determine his/her location using Advanced Symmetric key transfer on a public grid. The second step involves a private information retrieval interaction that retrieves the record with high communication efficiency. We analyzed the performance of our protocol and found it to be both computationally and communicational more efficient than the solution by Ghinita *et al.*, which is the most recent solution. We implemented a software prototype using a desktop machine and a mobile device. The software prototype demonstrates that our protocol is within practical limits.

Future work will involve windows mobile application and web application to implement multiple clouds testing the protocol on many different mobile devices. The mobile result we provide may be different than other mobile devices and software environments. Also, we need to reduce the overhead of the primality test used in the private information retrieval based protocol. Additionally, the problem concerning the LS supplying misleading data to the client is also interesting. Privacy preserving reputation techniques seem a suitable approach to address such problem. A possible solution could integrate methods from. Once suitable strong solutions exist for the general case, they can be easily integrated into our approach.

REFERENCES

1. R.Paulet,M.GolamKaosar,X.Yi,andE.Bertino,“Privacypreserving and content-protecting location based queries,” in Proc. ICDE, Washington, DC, USA, 2012, pp. 44–53
2. X. Chen and J. Pang, “Measuring query privacy in location-based services,” inProc. 2nd ACM CODASPY, San Antonio, TX, USA, 2012, pp. 49–60
3. B. Palanisamy and L. Liu, “MobiMix: Protecting location privacy with mix-zones over road networks,” inProc. ICDE, Hannover, Germany, 2011, pp. 494–505
4. L. Marconi, R. Pietro, B. Crispo, and M. Conti, “Time warp: How time affects privacy in LBSs,” in Proc. ICICS, Barcelona, Spain, 2010, pp. 325–339
5. G. Ghinita, C. R. Vicente, N. Shang, and E. Bertino, “Privacypreserving matching of spatial datasets with protection against background knowledge,” inProc. 18th SIGSPATIAL Int. Conf. GIS, 2010, pp. 3–12
6. Pingley, N. Zhang, X. Fu, H.-A. Choi, S. Subramaniam, and W. Zhao, “Protection of query privacy for continuous location based services,” in *IEEE INFOCOM, 2011.*
7. 7R. Dewri, I. Ray, I. Ray, and D. Whitley, “Query m-Invariance: Preventing query disclosures in continuous location-based services,” in *MDM*, 2010.
8. M. L. Yiu, G. Ghinita, C. S. Jensen, and P. Kalnis, “Enabling search services on outsourced private spatial data,” *VLDB Journal, vol. 19, no. 3,* pp. 363–384, 2010
9. W. K. Wong, D. W. Cheung, B. Kao, and N. Mamoulis, “Secure kNN computation on encrypted databases,” in *ACM SIGMOD, 2009.*
10. Khoshgozaran and C. Shahabi, “Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy,” in *SSTD,* 2007.