ISSN: 2319-6505

Available Online at http://journalijcar.org

International Journal of Current Advanced Research
Vol 5, Issue 8, pp 1180-1185, August 2016

International Journal of Current Advanced Research

ISSN: 2319 - 6475

RESEARCH ARTICLE

NOVEL TECHNIQUE TO PRESERVE PRIVACY AND OPTIMAL MEETING LOCATION FINDER ON ANDROID DEVICE

Nishant B.Chobitkar

IT Department SCOE, Vadgaon (BK) Pune, Maharashtra, India

A R T I C L E I N F O

Article History:

Received 24th May, 2016
Received in revised form 16thJune, 2016
Accepted 4th July, 2016
Published online 28th August, 2016

Key words:

Privacy, Android application, Optimal meeting location, Advanced Encryption Algorithm.

A B S T R A C T

Now a day's highly interlinked urban residents are increasingly dependent on smart phones and mobile devices to map and manage their daily lives. In modern data sharing society most of the people demands some extra mechanism to share their resources with the help of smart phones. This smart phone plays a huge role in it. These smart phones contains lots of applications to provide service to the user, location based services is including in this case. But the major question arises, that the sharing data is secure or not? For answering these everybody depends on the third party service providers. But many of the people do not want to reveal their location information to third party service provider. In this paper, we present an android application that give facility to calculate the optimal meeting location and time to meet without revealing their information to other user or third party vendor. Advanced Encryption Standard algorithm is used for privacy-preserving of meeting location data.

INTRODUCTION

The fast abundance of mobile phone technology in metropolitan communities has enabled mobile users to make use of situation based services on their devices. Service providers get benefit of this active and ever-growing technology landscape by proposing novel context-dependent services for mobile users. Location based Services (LBS), for instance, are employed by several mobile subscribers each day to get location-specific information. Location- based services (LBS),presented by such providers and used by all mobile user each day, have verified to be very effectual in this perspective[1]. Location check-in and its sharing are two trendy applications.

By monitoring into a place, subscriber shares their current location with their families or friends, and to those who do it regularly may also get particular deals, provided by the nearby businesses, as incentives for sharing their locations[2]. For instance, newly launched service by which users who want to check-in can come across on-the-spot discounts and deals. On the basis of the services provided by on location-sharing which is existing, almost 20-30% of mobile users are undoubtedly becoming popular. For instance, recently declared application that exploits location data from various users is a taxi-sharing application, offered by a global telecom company [3].In order to share a taxi, users have to disclose their departure and destination points to the server. Determining an appropriate location for a set of users is a pertinent concern.[4] Several existing supplier offers various of these services either using online web applications or with the help of separate applications for mobile devices. This features is desirable as well as it optimize for trade-off between ease and cost for the occupied parties [5], [6]. However, there are rising problem about how private data is used and work is performed on it by these providers. A study on privacy in location sharing- based services (LSBS) is made where with 25 participants (college students and non scientific personnel), and according to the results 80% of them consider it is significant to guard their location privacy from non-permitted users. Comparable results have been obtained in a study on location-based services (LBS) [1]. Without efficient safety, inadequate location information has been shown to offer steady details about a user's personal sphere, which has harsh effect on the user's public, private and economic existence. For example, a web service has revealed how thieves may mistreat users' location updates (from a well-known online social network) to rob their house while they are not at home [7].

In the taxi-sharing application, if the server is not fully trusted by all users, exposing sensitive locations (such as users home/work addresses) could be attacked by some third-parties. Thus, the revelation of location data to potentially untrusted third-parties and peers must be limited in any location-sharing-based service. In this paper, the privacy issues in LSBS has been addressed by studying one practical and related instance of such a basic state, which is the calculation of a fair rendezvous point (FRVP) in a privacy-preserving way, given a set of user-provided locations. This is a new and potentially important problem for LSBS applications, which holds the essence of the computations that are generally necessary in any LSBS, and mitigates their inherent and important privacy issues. We also present an accurate and absolute investigation of the privacy features of our proposal and explain that algorithms do not give any probabilistic advantage to a passive contestant in properly calculating the preferred location of any member.

Related Work

Igor Bilogrevic, Murtuza Jadliwala[1] planned privacy-preserving algorithms for calculating an optimal meeting location for a group of users. They execute a through privacy assessment by formally measuring privacy-loss of the planned methods. They address the privacy concern in LSBSs by focusing on a specific problem called Fair Rendez-Vous Point (FRVP) problem. Given a location preferences for set of users, the FRVP problem is help to find out a location among the proposed ones such that the largest distance among this location and all remaining users' locations is reduced [1].

LinkeGuo, Chi Zhang proposes a privacy-preserving revocable substance sharing system in geosocial networks. Proposed scheme helps mobile users to share their encrypted location-based contents on an untrusted server without exposing original detail of location, and enables other users of mobile device who actually verify at the meticulous location to find and decrypt the matter if they have the corresponding attributes [8].

Pidcock proposes to disassociate user identity information from user location information in our privacy-friendly location hub. No person should be familiar with both a user's identity and user's location. The groundwork of location hub, ZeroSquare, is two noncolluding entities, one that supplies information about users and another that supplies information about locations. ZeroSquare also provides a callback framework to bear scenarios where a user wants to be notified when a condition is met. However, by having only users (but not locations) become first-class citizens in the architecture, the applicability of these architectures to geosocial applications remains partial because accumulating or recovering information about locations is hard [9].

Guha proposed a privacy-preserving system. The main aim of system is that rather than sharing the location it allows user to set location triggered alarms based on presence at specific location [10].

Berger proposed a skilled meeting-location algorithm that considers the time in-between two successive meetings. However, all data about users is public [11].

Santos and Vaughn [12] present a survey of existing literature on meeting-location algorithms and propose a more comprehensive solution for such a problem. Although considering aspects such as user preferences and constraints, their work (or the surveyed papers) does not address any security or privacy issues.

Frikken and Atallah [13] propose SMC protocols for securely computing the distance between a point and a line segment, the distance between two moving points and the distance between two line segments.

Proposed System Architecture

In addition to the study, we also calculate the practical usefulness and performance of the proposed algorithms by means of implementation on Android mobile devices. We also deal with the multi-preference case, where each user may have more than one preferred location preferences. We underline the major differences, in terms of privacy and performance, with the single preference case, and also give trial results for the multiple-preference execution. Finally, throughout the user study, we provide usability of proposed solution. We provide our contributions are as follows. First, we describe the results of our focused user-study on location-sharing and privacy in mobile devices. Secondly, motivated by the results of this study and the need for privacy in LSBSs, we propose and study practical solutions to the FRVP problem, which do not reveal any extra information to third parties or further peers. The proposed solutions are autonomous of any fundamental service or network provider, and can be included in existing location-sharing-based services. Third, we calculate the strength and flexibility of our schemes to both passive and active attacks through a privacy investigation of the proposed result. Fourth, by implementing our proposed algorithms on a test bed of real mobile devices, we illustrate that their performance in computing the rendezvous point is satisfactory, and that users do not incur in vital bonus transparency due to the inherent privacy features.

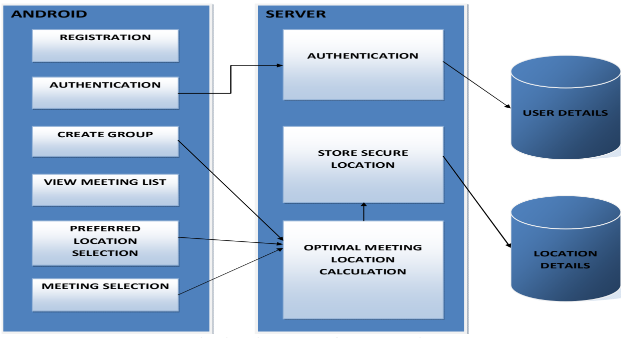

Fig 1 Block Diagram of Proposed System

Proposed system working

The main goal is to spot the location of the member and to arrange meeting, which provides all the members in the group. The location of the members is tracked using the GPS in mobile phones and it gets stored in the storage. Based on the current or preferred location of the members common place is known using the Google maps and ADA is used to calculate central location. That exact meeting location and address will be sending to the corresponding member's mail-id in encrypted form.

Working modules in android

Android application doing all input operation and give facility for getting location of meeting. Below are the stages of Android application. Facilities in android application like registration, create group, arrange meeting, and view optimal location.

Stage 1: Registration and login

User can register them using their mobile from the application and can login with their credentials. User can register them using following details as follows:

- Name
- Username
- Password
- Email id
- Address
- Mobile number

With the use of username and password of registered user login can be done.

UI design and layout

Android device are programmed in Java. Layouts are use for organization. The graphical layout can be decked with Text field, button, scrollbar, etc. The designing of interface for user can be done in HTML and decorated in CSS for good performance.

Stage 2: Authentication

Registered id will be taken for login, and password verification will also be done during login. When user going to register himself he will get error message, if he type wrong email id, username and password. At the time of login if user tries to write wrong username or password, then he will get wrong username and password popup.

Stage 3: Create group

Every user can search other users using their username and can add them in their list of group zone. Group zone are set of friend list created by each user. After login, any user will take initiative to create group of users. Those users can be select in a group who want to attend the meeting.

Selecting the Users and Getting the Common Places

The admin will select users for group creation. Again admin will select that group of users from database and their current or preferred location can be tracked. The optimal meeting location place can be calculated and shows on Google Map.

Stage 4: Preferred location selection

The user will be given option to select preferred location 6 hours earlier to meeting time, If not selected will be removed from group. Those user who can be in group need to select one location from given list. This location can be given to admin. From preferred or current location of member optimal venue can be calculated.

Getting position from GPS receiver

With the help of Android Location Package and Google Map library classes, map and location based abilities can be used in application.

Stage 5: Meeting Selection

User can select the number of users from their group list with time and can proceed to meet them. Selected users are those who wants to attend the meeting and hence the selection should be done carefully else it may leak out the meeting location

View Pending Meetings List

In this section application will display details of pending meetings and give option to select preferred location for meeting.

Stage 6: View Optimal Location

Finally in application user shows the decrypted optimal meeting location in Google map. This location is the final meeting location of group.

Working Modules in server

In server side all operation on user data, user authentication will be done.

Stage 1: Authentication

In server side, if the credentials match, the process is completed and the user is granted authorization for access. Credentials like username and password. User inserts his username or password then server checks the details in database and gives message to the user either “Wrong Username or Password” or “Login successfully”.

Sending data from mobile phone

Data received on application like username, password, latitude, longitude can be sent to the server. All parameters are encoded and send to the server using android application

Receiving data from mobile phone

The server receives all the parameters from the mobile phones and then connects to a database and executes a series of queries to save all the parameters in their individual tables.

Stage 2: Optimal meeting location Calculation

Based on the preferred location selected by the user, application will calculate the optimum location of meeting considering the location of each attendee. Application will calculate the shortest path for every individual and a suitable location will be termed as a meeting location**.**

Aggregate Distance Analysis (ADA)

```
Identify selected users and their corresponding locations
Identify closest co-ordinate to all selected locations
Identify latitude and longitude of each user location
for each location
    closest_location_latitude += latitude
    closest_location_longitude += longitude
end for
closest_location_latitude = closest_location_latitude / no of users
closest_location_longitude = closest_location_longitude / no of users
Identify list of locations tracked for meeting and their coordinates as locations
initialize min_distance = location_distance of first location with closest_location
for location[i] in locations
    if min_distance > location[i] distance with closest_location then
        min_distance = location[i] distance with closest_location
    end if
end for
Select location with min_distance and display to user
```

We have to identify selected users and their corresponding locations. So, we find out closest co-ordinate to all selected locations and also Identify latitude and longitude of each user location for each location. We assign closest location latitude and longitude .the optimal location latitude assign to optimal location latitude divided by no. of users and optimal location longitude assign to optimal location longitude divided by no. of users. Then identify list of nearby locations from calculated optimal location and their coordinates as locations. We initialize min distance assign to location distance of first location with optimal location. If min distance is greater than location distance with optimal location then min distance is equal to location distance with optimal location. Then Select location with min distance and display to user.

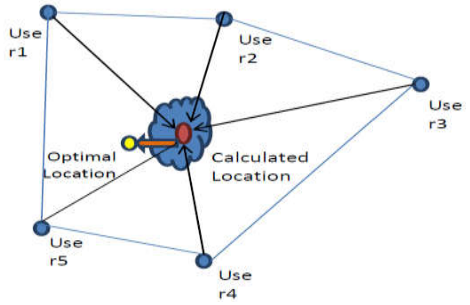

Fig 2 Proposed System Process Diagram

Fig shows the working process of proposed system. This process comprises of multiple stages of execution. As per shown consider a condition of the existence of the five users in group planning to meet in centrally preferred location then one user from all will become admin and after which all user will share their location with admin and the admin then executes the entire process. After execution system will calculate the central location by calculating the centroid of the polygon generated by the user's connection. Once system get the central location it will ask user about his preferred location and after this using Google mapping API system will find out the nearest location selected by the user and once it is found system will inform all the remaining user about final meeting location. If user wants he can view the travelling path to the location.

Mathematical Model

System S= {Optimal Meeting Location Application}
System S= {S1, I, D, O}
S1 = {User, Location}
L1={Closest Location latitude}
L2={Closest Location longitude}
OL={OL1,OL2}
OL1={Optimal Location latitude}
OL2={Optimal Location longitude}
I = {Coordinates}
D = function
O = output

1. D1=Identify(latitude, longitude)
2. Location= for each
   L1={Closest Location latitude}
   L2={Closest Location longitude}
3. OL1=OL1/no. of users
4. OL2=OL2/no. of users
5. D2=Identify(list of locations)
6. D3=min(distance of first location, OL)
7. Location=Location[i]
8. D4=min(distance) > Location[i]
9. D5=min(distance) = Location[i]
10. D6=Select(Location_min(distance))
11. O=Display(Location_min(distance))

Stage 3: Store secure Location

After the calculation of optimal meeting location, location data can be store in location database in encrypted form with the help of AES mechanism. Store location data are ready to send on user application. All data can be sending to user in encrypted form. In server side, server sends data to respective user id to store.

Stage 4: Privacy using encryption

The location data is encrypted and then passed on to each individual concern with the meeting. Also the decryption key of the data is mailed to each individual so that they can use the key to decrypt the data and get the location of meeting. With help of the key, privacy of the data is been stored and the location can be accessed by people who are related with meeting.

Advanced Encryption Standard (AES) Steps:
1. Encryption of calculated location using key and cipher
2. Generate secret key
3. Output stored in database
4. Using secret key user will get location
5. Final optimal location

L= Optimal Meeting Location
C= Cipher
K= Key
SK= Secret Key
E_l=Encrypted Optimal Meeting Location
Step 1: Calculated optimal meeting Location can be encrypted using key and cipher. In encryption scheme location data (plain text) can be converted to cipher text and generate secret key.
E_l= L || K || C
Step 2: Generated secret key and encrypted location then send to user by mail.
Mail Data= E_l || SK
Step 3: Server performs an action of storing encrypted data into the database as well.
Database= E_l || Group
Step 4: With the help of secret key (i.e. sent on email) user can decrypt their location for meeting.
D_l= SK || E_l
Step 5: After decryption user will get accurate as it is location data on Google map.
D_l= L

Sending the Meeting Location via Mail

The scheduled meeting contain meeting name, venue, time, date, will be send to registered email id.

Location plotting on Google Maps

The webpage contain inbuilt Google Maps using Google JavaScript API V3.API key can be used for loading of Maps API using Maps API application. All maps API applications should load the Maps API using an API key. The key is embedded in Google maps. Latitude and longitude coordinates are received and are plotted.

Working Modules in Database

All data related to user like name, username, password, location data and final optimal location in encrypted form can be stored. Mysql will be use for performing action on database. Two database will be create as follows: (1) User Details (2) Location Details

Stage 1: User Details

In this database, users all credentials have been stored and connected to the android application for getting input data with the help of server. In server, authentication will be check whether the input entry is belonging to that user details or not. In practical point of view below steps has been performed by application and stored that data.

Step 1: Registration details of user stored into database.
Step 2: Username and password stored to their respective user.
Step 3: Created group entry with its name, time and date

Stage 2: Location Details

Meaning of this database is self-explanatory, means database contain all location details. This database contain following details:

- Users current or preferred location
- Optimal meeting location in encrypted form

Performance Evaluation

Computation Delay on Android Device

As it can be seen, ADA and AES algorithm is the most proficient for the distance computations, require only 1 second for 2 users. In general, we can see that the ADA and AES algorithm has a better performance for computing optimal meeting location. Moreover, the ADA and AES based system is the most capable across all computation, require only 4 seconds executing the protocol with 10 participants. For getting computation delay of system we are plotting graph with estimated time for computation against number of users.

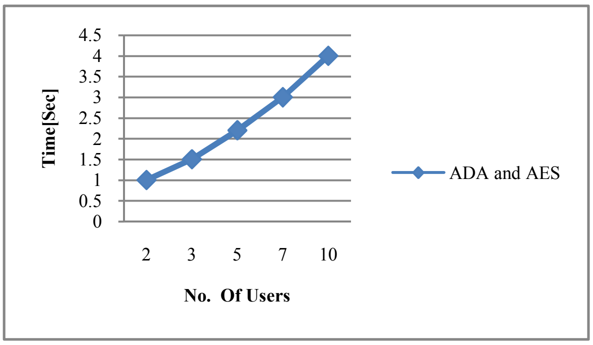

Fig 3 Computation Delay on Android Device

Comparison of Proposed and Existing System

Fig below shows comparison between existing and proposed system. In existing system Elgamal-paillier algorithm can be used for computing optimal location.

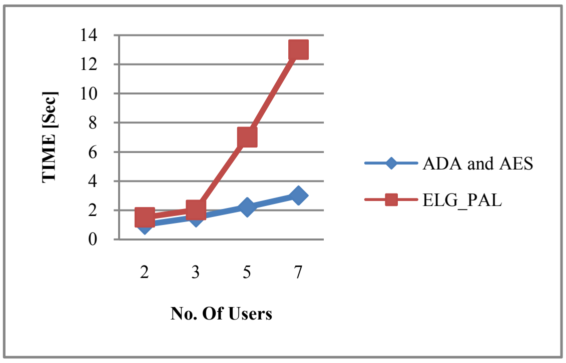

Fig 4 Comparison of Existing and Proposed System

In proposed system Aggregate distance analysis (ADA) and Advanced Encryption Standard (AES) can be used for getting accurate optimal location. Below graph clearly shows that proposed system has better performance than existing system for processing time. Proposed system is preferred over existing system for large number of users in group. For 7 users in group Elgamal-paillier required 13 sec for calculating optimal location in our proposed system it required only approx 2.3 sec. It clearly shows that proposed system has great performance.

RESULTS

We used personal system consisting i5 processor with 2.4 GHz 3GB RAM. The storage of HD is 500GB. Hence are the hardware specifications and the operating system we used was Win7. The editor we used was IDE-Eclipse. Version of JAVA was JDK 1.6.0. The database for synchronization was MYSQL. Specifications mentioned are the minimum individuals; maximum are suggested for better issues avoidance.

The corresponding figure shows the overall working environment for all of the following modules. Hence the design shows starting from the user registration and login, create group, arrange meeting and final optimal location on Google map, thus by scheduling the accordance meeting and notifying the selected users.

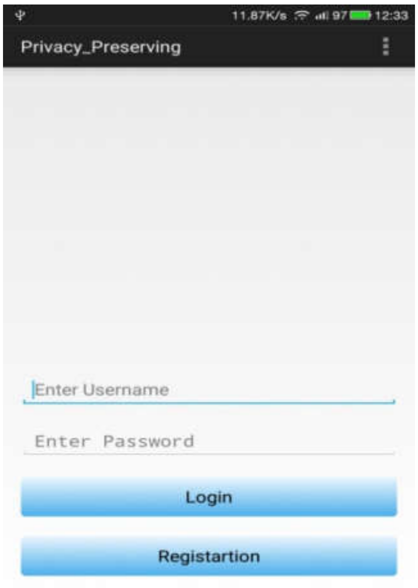

Fig 5 User Login

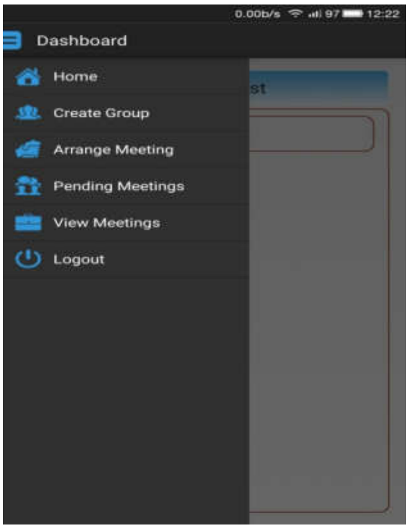

Fig 6 Application GUI

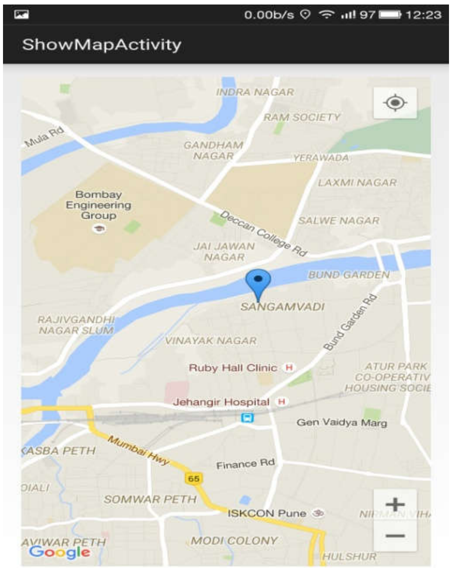

Fig 7 Optimal location on Google map

CONCLUSION

The Privacy Issue in optimal meeting location calculation is addressed genuinely. The security and privacy procedures are handled by well-known cryptographic theory like AES. This system experimentally demonstrates that the solutions preserve user preference privacy and have satisfactory performance in a real implementation. Moreover, the proposed approach is extended by algorithms to contain cases where users have several prefered locations. Finally, based on an extensive user study, this approach showed that the proposed privacy features are crucial for the adoption of any location sharing or location-based applications.

References

1. Igor Bilogrevic, Murtuza Jadliwala, Vishal Joneja, "Privacy-Preserving Optimal Meeting Location Determination on Mobile Devices", IEEE Transactions on Information Forensics and Security , Vol. 9, NO. 7,2014.
2. E. Valavanis, C. Ververidis, M. Vazirgianis, G. C. Polyzos, and K. Norvag, "MobiShare: Sharing context-dependent data & services from mobile sources," in *Proc. IEEE/WIC Int. Conf. WI*, Oct. 2003, pp. 263–270.
3. (2011, Nov.). *Orange Taxi Sharing App* [Online].Available:http://event.orange.com/default/EN/all/mondial
4. P. Golle and K. Partridge, "On the anonymity of home/work location pairs," in *Proc. 7th Int. Conf. Pervasive Computing*, 2009, pp. 390–397.
5. J. Freudiger, R. Shokri, and J.-P. Hubaux, "Evaluating the privacy risk of location-based services," in *Proc. 15th Int. Conf. Financial*, 2011,pp. 31–46.
6. J. Freudiger, M. Jadliwala, J.-P. Hubaux, V. Niemi, P. Ginzboorg, and I. Aad, "Privacy of community pseudonyms in wireless peer-to-peer networks," *Mobile Netw. Appl.*, vol. 18, no. 3, pp. 413–428, 2012.
7. J. Krumm, "A survey of computational location privacy," *Personal Ubiquitous Comput.*, vol. 13, no. 6, pp. 391–399, 2009.
8. LinkeGuo, Chi Zhang, "Privacy-Preserving Revocable Content Sharing in Geosocial Networks", IEEE Conference on Communication and Network Security, 2013.
9. S. Pidcock and U. Hengartner, "Zerosquare: A privacy-friendly location hub for geosocial applications," in *Proc. 2nd ACM SIGCOMM Workshop Networking, Systems, and Applications Mobile Handhelds*, 2013.
10. S. Guha, M. Jain, and V. Padmanabhan, "Koi: A location-privacy platform for smartphone apps," in *Proc. 9th USENIX Conf. NSDI*, 2012.
11. F. Berger, R. Klein, D. Nussbaum, J.-R. Sack, and J. Yi, "A meeting scheduling problem respecting time and space," *GeoInformatica*, vol. 13, no. 4, pp. 453–481, 2009.
12. P. Santos and H. Vaughn, "Where shall we meet? Proposing optimal locations for meetings," in *Proc. MapISNet*, 2007.
13. K. B. Frikken and M. J. Atallah, "Privacy preserving route planning,"in *Proc. ACM WPES*, 2004, pp. 8–15.

****