Abstract

Background/Objectives: We propose a cloud based computing architecture to preserve data and privacy of the user’s location information and the interests of different intruders on service user’s data. Methods/Statistical Analysis: Data confidentiality and trust is in spike when it comes to modern cloud computing as storage and means of communication. Social Networking, usage of Location based services; cloud services offered by CSPs have become part of one’s daily life with the advent of smart phones. Findings: Location information of the user can be obtained using GPS (Global Positioning System) or RFID (Radio Frequency Identification). Hacking which the access to the data of the user is obtained. This forms two limitations, 1. Works only in GPS enabled devices. 2. If mobile devices perform such heavy computations it leads to battery depletion problem. To overcome the second limitation Mobile Cloud Computing is used. Improvements/Applications: Mobile Cloud Computing offers great computing power and storage to the mobile devices. It also increases the affordability and reliability but it is still in its budding stages in providing trust guarantees to the user.

Keywords: CSP (Cloud Service Provider), Data Confidentiality, Location Based Services, Location Privacy, Multi-Level Access Control, Mobile Cloud Computing

1. Introduction

Location Based Services (LBS) applications have massive demand worldwide due to the advent of Smartphone’s, equipped with GPS modules and powerful computation ability. Hence the mobile devices are used not only for communication purpose but also to make use of rich collection of applications available such as Google apps and iPhone apps. While obtaining services from these applications most of the mobile devices face problems regarding security, storage, battery life and bandwidth due to their limited hardware capabilities. To overcome these problems developers started to adopt mobile cloud computing (integrating mobile devices with cloud computing).

In early 2000’s LBS was mostly been used in commerce with a subscription based business model. But now LBS is used to provide services such as health, business, tracking, information, emergency and advertising services. There are also applications that link users with face book to find the nearby events and other social networking related applications. These applications offer attractive services to the user and has shown a new way of utilizing mobile phones. The location of the user can be detected using Global Positioning System (GPS) and Radio Frequency Identification (RFID). Pew Internet and American life project research found that around 74% of Smartphone users use location based services. The research has also found that most of the mobile users have turned off their location tracking features at some point due to privacy concern or heavy battery consumption of these applications.

LBS provide services to the user based on the exact geographic location of the user. When the user sends his/her current location co-ordinates to obtain the service, the hacker can trace the user activities and harass the user using the location information. For example when X and Y users meet for an official cause the observer can take their private information and disclose that they are in relationship and hence harasses the user. He can also link the user activities (by finding his past and present location) and can deduce private information.

In tracking services most of the applications give location information of the user based on the group the
user has specified i.e. which group can or cannot access the exact location.

In this paper we propose LBS-TI architecture which combines the two unique features of LBS, 1. Friend Locator (Tracking services) and 2. Finding location associated services (information services). Our platform uses LBS based on the idea of cloud server. The Key Generator and Interface (KGI) server is intermediate between a user and LBS Service provider. A privacy preserving protocol will be used to query and to get location information (in friend locator) without violating user location privacy. The location information will be given to the friend in an access controlled manner based on the relationship with the other user.

2. Literature Survey

Several approaches are already been presented to preserve the privacy of the users. These approaches can be broadly classified into 3 types as follows, 1. Anonymity, 2. Transformation, 3. Private Information Retrieval (PIR).

Anonymity refers to a state of being unidentifiable within set/group of users i.e. to form a group of ‘K’ users and remain anonymous among them. These K users can either be real or dummies in the cloaked regions. The cloaked region can be formed in either a centralized or decentralized manner depending on where the cloaking takes place. K-anonymity model is a centralized clocking technique which provides option to the user to specify the minimum level of anonymity as well as the maximum temporal and spatial resolution. But the drawbacks here are 1. Anonymizers have to be trusted. 2. Centralized anonymizer should be more sophisticated. 3. If the repeated requests of a user are observed then he/she can be distinguished easily among K users.

To overcome disadvantages of the centralized anonymizer, Generation of dummies a decentralized method was proposed. In this technique, the user sends true position information along with several pseudo location information called as dummies to Service Providers (SP). The SP cannot know the true position information and hence privacy of the user is preserved. But the drawback is slow server response due to the increasing number of requests sent by the user. The dummies must be selected intelligently else valid user information can be revealed easily. The Mixed zone model anonymizes user identity by restricting the positions where users can be located. It aims to prevent tracking of long term user movements. To provide anonymity many pseudo paths will be generated based on ingress and egress events of the user. But this technique is computationally expensive.

PIR enables the user to send a query to LBS server without revealing the request. PIR approaches can be further divided into theoretical and hardware based (Practical) approaches. The techniques in uses theoretical and practical PIR techniques respectively. These approaches provide faultless privacy guarantee to the user but it faces severe problems like hardware reliance, computational complexity and communication cost.

Transformation is based on transforming a query in secure manner such that the server is unable to identify the user’s location. It is an emerging approach which tries to alleviate some of the weaknesses of the anonymity model. There are two types of transformations, 1. Spatial transformation. 2. Non-Spatial transformation. Spatial transformation is based on geometric transformation (scaling, transformation and rotation) where as non-Spatial transformation is based on the cryptographic principles. The technique proposed by Len is based on spatial transformation method. In this method location privacy is enabled by interposing agents between users and the SP to perform the transformations. The agents do not store user information, they act as intermediaries. The only responsibility of agents is to transform information received from other users or the server. Users should choose the agents randomly to preserve privacy and to perform the transformation.

Yiu et al. proposed a framework called Space Twist which blinds the untrusted location server by incrementally retrieving Points Of Interest (POI), based on the ascending distance of POI from a fake location which is near the query point (termed as anchor point). In this method, the query is evaluated in the original space but the query point is transformed to an anchor point. Transformation eliminates the need for trusted third party during query processing and hence it is advantageous over anonymity model. It also provides efficient query processing with privacy. It has the following disadvantages: it becomes costly in computation or communication wise if the exact results and the privacy are required at once.

3. Government Interception with users Data

There are various ways the government can have access to the user’s data. It also varies worldwide. Some country
enforces data confidentiality, while other promises and other do not take it seriously restricting it to only for military and terrorism. The common three ways used by government to access user’s data are

- **SLA violation regarding storage encryption**
  The CSP encrypts the data stored in the server in his data center. This technology makes the cloud storage data unique with respect to SSL, https or TLS. Here the key can be kept private with the user or it can be owned by the CSP or both. In this the Service provider can decrypt the data and the information can be leaked. Hence the Key can be kept with the user itself and users datas will not be in stake. Example Hushmail a US based email service used by US government to know the suspected drug dealers communication and distribution centers.

  Solution: Service providers should turn on https/TLS as it is highly efficient and error is less than 1% or the key should only be owned by the user.

- **SSL Interception assault**
  SSL relies on the public key infrastructure involving a chain of trust built up of several Certificates Authorities (CA) in a hierarchy to validate a websites certificate. Here the Government takes into control any of the CA in any level and issues fake certificate, by which they are able to get all the information, data through the URL logged in by the user. Example Packet Forensics 5 series device which can be plugged at any intermediate level or root to issue fake certificates.

  Solution: A firefox plugin called CertLock can be used which reveals the user that fake certificate is issued as CertLock which has the certificate copy of the first visit made by the user and for each login, it maps with the original. Hence information from other country access or any intruder will be revealed to the user.

- **Unobtrusive software changes**
  The service provider may change the software version backdoor implicitly for all the users or specific for a set or single user when forced by government, where the changes in the version is not maintained by the CSP and also left unnoticed by the user. Hence a threat to data confidentiality arises when backdoor software’s are installed to retrieve the datas.

  Example The German Government used Java Anonymous Proxy to trap down access of the IP addresses to illegal websites.

In the LBS-TI approach various cryptographic concepts are used to provide privacy to the LBS users. We introduce each of them in this section.

### 3.1 Paillier Homomorphic Cryptosystem

It is an asymmetric type of cryptosystem which allows specific types of computations to be carried out on ciphertext. It is composed of three algorithms 1. Key Generation. 2. Encryption and 2. Decryption.

  **KeyGeneration:** The user randomly chooses two large prime numbers p and q of same bit-length. He then computes n = pq and λ = (p – 1)(q – 1) and calculates g and μ as g = (n + 1) and μ = (λ mod n°)°1 mod n. The encryption key, EK = (n, g) and the decryption key, DK = (λ, μ).

  Encryption and Decryption: Encryptor then selects a random integer r € Z and computes cipher text as C = g°M°r°n mod n°2 where M is the plain text. To recover the original message decryptorcomputes the following M = L(C°λ°mod n°2). μ Mod n, where L(a) = (a – 1)/2.

### 3.2 Cipher-Text Policy Attribute based Encryption (CP-ABE)

It is a type of identity based asymmetric encryption which as one public key and one master private key from which other private keys are generated. It is a flexible encryption method that allows complex rules of specifying which private keys can decrypt which ciphertexts. The private keys are associated with a set of attributes and ciphertext specifies an access control policy over these attributes within the system. A user can decrypt the ciphertext if and only if his attributes satisfy the access control policy of the ciphertext. Here authorization is included as a part of encrypted data.

In most of the ABE works encryption policy is described along with the access tree. In our approach we store the level of location disclosure in the leaf nodes and conditions that satisfy to obtain the location information as non-leaf nodes of the access tree.

### 3.3 Secured Hash Algorithm (SHA)-512

SHA is a cryptographic hash function that uses mathematical operations on electronic data. One can compare the known hash value with the computed hash value to determine data integrity. The algorithm takes message of length <2°128 bits as input and produces a message digest of length 512 bit as output. The input is processed in 1024 bit blocks. The overall processing of the
message to produce a message digest consists of five steps as follows 1. Appending padding bits. 2. Append length. 3. Initialize hash buffer. 4. Process message in 1024 bit blocks – It has 70 rounds. 5. Output. The algorithm has a property that every bit of the hash code is a function of every bit of the input.

3.4 Data Encryption Standard (DES)

DES is a predominant symmetric encryption algorithm. The structure of the DES is slight variation of Fiestal Cipher structure. It is a block cipher method which processes plain text as a 64 bit blocks. The key is 56 bits in length from which 16 sub keys are generated one for each round. Encryption and Decryption process are same but the keys should be used in reverse order in the decryption process i.e. \( k_{16} \) for the first round, \( k_{15} \) for the second round and so on until \( k_{1} \) is used for the last(16th) round.

3.5 Elliptic Curve Cryptography (ECC)

Elliptic Curve Cryptography (ECC) is a public-key cryptography approach which is based on the algebraic structure of elliptic curves over finite fields. One of the main benefits is that the security is provided for keys of smaller size. It is also used for encryption, digital signatures, pseudo-random generators etc. It is a variant of RSA cryptosystem.

4. Proposed Architecture

We propose a cloud server based LBS-TI architecture for location based services. Most important components of our architecture are the Key Generator and Interface server which mediate between users and LBS servers. The reason behind choosing cloud based server is to exploit the computing facilities offered by cloud along with processing power, reliability and scalability. The location information of the user is not stored in either of these servers hence the attacker will not get any useful information from this data. The architecture uses four encryption algorithms (as discussed in the previous section) for communication.

Figure 1 shows the proposed architecture. During initialization the user needs to register with the KGI server to obtain the Location Based Services. Locating a friend (tracking service) and 2. Location associated information service such as to find the nearby hospitals, restaurants, post offices, etc. Once a user is registered with the KGI server, it issues keys to the user for further communications.

If the user \( u_i \) selects locating a friend service, then the user \( u_i \) should send a query to the friend (say \( u_j \)). The \( u_j \) (friend) can specify access control policy like a user can know my exact location if s/he is the family member or a user can know the exact distance between them if s/he is the classmate or a user can give approximate distance between them if s/he is from the friend list. This access control policy is specified as an access tree ‘T’ on all possible attributes of the user. The location request message from the user \( u_i \) will consists of current location information \( x_i \) of the user \( u_i \) and a set of attributes \( A_i \) that determines \( u_i \)’s identity. If \( u_i \) is in the friend list of \( u_j \) then s/he will be identified as an authorized user and then the request will be processed further to know the relationship of the user \( U_i \) with \( U_j \) and executes a function \( f \) on \( u_j \)’s location information \( x_j \). And accordingly \( u_i \) will obtain the query result as \( f(x_i,A_i,x_j) \). The (\( x_i,A_i \)) pair leads to different level of query results. \( u_i \) cannot know any other information of \( u_j \) apart from the location information he has access to. Paillier and CP-ABE encryption algorithms are used to provide multilevel access control to the user.

Where as if the user chooses information service then the request message consist of \( u_i \)’s current location information \( x_i \) and name of the service (like hospital, restaurant, library) \( S_i \). \( u_i \) is searching in for at that location. This request will be forwarded to the KGI server which in turns forwards it to the LBS server. LBS will query...
its database and prepare a response message and send it back to the KGI which in turn forwards to the user. The response message consists of a list of available services at location $x_i$. Here, SHA-512 and DES encryption algorithms are used to ensure data integrity.

5. Implementation

At the time of Initialization users need to register themselves with the KGI server; on successful registration the server distributes keys to the user for further transactions. The application provides two types of services to the user:

- Tracking Service.
- Location associated Information Service.

The users can choose either of the services and get the required information.

5.1 Tracking Service

When the user (say $u_i$) selects locating a friend service, then the user $u_i$ can send a query to his friend (say $u_j$) directly without any server intrusion. The $u_i$ (friend) can specify access control policy (conditions) on the location he shares with his friend based on the relationship, like a user can know exact location if s/he is a family member or a user can know the exact distance between them if s/he is a classmate or a user can know whether s/he is in the threshold range if s/he is a friend.

The location request message from the user $u_i$ will consists of current location information $x_i$ of the user $u_i$ and a set of attributes $A_i$ that determines $u_i$’s identity. If $u_i$ is in the friend list of $u_j$ then he will be identified as an authorized user and then the request will be processed further to know the relationship of the user $U_i$ with $U_j$ and then executes a function ‘$f$’ on $u_j$’s location information $x_j$. Accordingly $u_i$ will obtain the query result as $f(x_i,A_i,x_j)$. The $(x_i,A_i)$ pair leads to different level of query results. User $u_i$ cannot know any other information of $u_j$ apart from the location information he has access to.

The access control policy is specified as an access tree ‘$T$’ on all possible attributes of the user. The conditions are placed in the non-leafy nodes whereas location information is stored in the leafy nodes. Paillier and ABE encryption algorithms are used to provide multilevel access control to the user.

5.1.1 Requestor (User1) Side

Begin()
Select friend()
Encrypt the request()
Send to responder()
.
.
Receive Response from the friend()
Decrypt and take info of friend location()
End

5.1.2 Responder (User 2) Side

Begin()
Receive request from Requestor()
Authenticate Requestor()
if (authorized_user)
Check AccessPermission()
if (access_permission)
Encrypt location info of responder()
Send to requestor()
else
Send warning()
Endif
Else
Send Access_Denied_Info()
Endif
End

5.1.3 Relationship based Location Access

As said earlier, based on the relationship of the responder with requestor, requestor can access different location information of the responder. Three types of relationship are considered here, 1. Friend, 2. Close Friend and 3. Relative.

Query 1: Friend

if (Friend)
Calculate Distance between requestor and responder()
Compare with threshold_value()
Encrypt resulted information()
Send to requestor()
Endif

The response message contains information as whether the user is within his threshold value or not. No other location information will be sent.
Query 2: Close Friend

if (Close Friend)
  Calculate Distance between requestor and responder()
  Encrypt resulted information()
  Send to requestor()
Endif

The response message contains exact distance between the requestor and responder. No other location information will be sent.

Query 3: Relative

if (Relative)
  Re-verify the user()
  Fetch current location()
  Encrypt resulted information()
  Send to requestor()
Endif

The response message contains exact location of the responder. Before giving the exact location information the user i.e. requestor will be re-verified to preserve data confidentiality.

5.1.3.1 Location Distance Computation

The haversine formula is an important equation in navigation, giving great-circle distances between two points on a sphere from their longitudes and latitudes. For any two points on a sphere, the haversine of the central angle between them is given by,

\[
haversin \left( \frac{d}{r} \right) = haversin(\epsilon_1 - \epsilon_2) + \cos(\epsilon_1) \cos(\epsilon_2) haversin(\lambda_1 - \lambda_2)
\]

where
- \( \epsilon_1, \epsilon_2 \): latitude of point 1 and latitude of point 2
- \( \lambda_1, \lambda_2 \): longitude of point 1 and longitude of point 2

5.1.3.2 Information Service

When the user chooses information service, the user can select type of service he is interested in at a particular location and send a request message to KGI Server. It consists of ui’s identity, interested location information \( x_i \) and name of the service (like hospital, restaurant, library) \( S_i \) is searching for at that location. The KGI server decrypts the message and in turn forwards it to the LBS server. The forwarded message contains only location and type of service. LBS will query its database and prepares a response message and sends it back to the KGI which in turn forwards to the user. The response message consists of a list of available services at location \( x_i \). Here, SHA-512 and DES encryption algorithms are used to ensure data integrity.

5.1.3.2.1 Requestor (User) Side

Begin()
  Select location()
  Select TypeOfService()
  Encrypt request msg
  Send request msg to KGI()
  .
  .
  Receive the requested service()
End

5.1.3.2.2 Responder (KGI Server) Side

Begin()
  Receive request()
  Decrypt request()
  Forward to LBS()
  .
  .
  Response from LBS()
  Forward to requestor()
End

6. Parameters Considered

6.1 Data Confidentiality

- Once a message is prepared, size of that particular message will be calculated and will be appended at the end and the message will be encrypted. If the attacker injects any data, the encrypted message will not change on the sender side and hence at the receiver side the injected message will dropped and he receives only the original encrypted message which he can decrypt it without any change..
- It makes use of several cryptographic concepts as discussed in the above sections to provide data security of location information of the user.
6.2 Privacy
- User details are not stored directly in the KGI Server, instead their hash values are stored and hence if KGI compromises also, attacker cannot obtain any useful information.
- KGI doesn't just forward the information service request of the user. It trims the user details from the request message and forwards request to LBS server. Hence if LBS compromise also, the attacker cannot obtain any useful information.

6.3 Authentication
- As mentioned earlier digital signatures are used to validate the user.
- In tracking service before giving any location details to a friend, he will be authenticated if s/he is in the list.

6.4 Reliability
- The LBS-IT architecture doesn’t disclose location details of the user in any scenario and hence the architecture is highly reliable.

7. Conclusion
Access control policy ensures that the user location information is not revealed to others unless it is intended by the user. Users who are related as “friend” knows the whether his friend is within the threshold region or not. Users those related as “close friend” knows the distance between himself and his friend. Whereas users who are related as “relative” knows the exact location of the user. Before giving the exact location of the responder, requestor will be re-verified.

Elliptic Curve Cryptography (ECC) is used during key distribution process. Location Information of the user is doubly encrypted for which efficient cryptographic algorithms like Paillier and Attribute Based Encryption (ABE) are used to provide multilevel access-control in tracking service and SHA-512 is used to verify signatures of the user as well as to store values in the database of KGI. Signature will be re-encrypted using DES encryption method in Information Service. The use of these algorithms delays the process of obtaining the LBS services in real time.

All the above mentioned concepts help to preserve privacy of the LBS user’s location Information.

8. Future Scope
The following features will be added and enhanced:
- More places can be added to the location dataset to make location displayed exactly.
- Currently the access-control policy is based on the relationship constraint and is fixed. In future a user defined constraint will be used to provide location details.
- Route maps and reviews for the selected location based service can be made available.

9. References
1. Location-Based Service (LBS) definition. 01 Jan 2015. Available From: http://searchnetworking.techtarget.com/definition/location-based-service-LBS
2. Location. 01 Jan 2015. Available From: http://www.pewinternet.org/files/old-media/Files/Reports/2013/PIP_Location-based%20services%202013.pdf
3. Abbas F, Hussain R, Son J, Oh H. Privacy Preserving Cloud based Computing Platform (PPCCP) for using Location Based Services. IEEE/ACM 6th International Conference on Utility and Cloud Computing; 2013. p. 103-18
4. Gedik B, Liu L. Location privacy in mobile systems: A personalized anonymization model. Proceedings of the 25th IEEE International Conference on Distributed Computing Systems; 2005. p. 620-9.
5. Kido H, Yanagisawa Y, Satoh T. Protection of location privacy using dummies for location-based services. Proceedings of the 21st International Conference on Data Engineering; 2005. p. 88-97
6. Beresford AR, Stajano F. Mix zones: User privacy in location-aware services. Proceedings of the 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops; 2004. p. 127.
7. Ghinita G, Kalnis P, Khoshgozaran A, Shahabi C, Tan K. Private queries in location based service anonymizers are not necessary. ACM SIGMOD International Conference on Management of Data; 2008. p. 121-32.
8. Khoshgozaran A, Shirani-Mehr H, Shahabi C. SPIRAL: A scalable private information retrieval approach to location privacy. 9th IEEE International Workshop on Mobile Data Management Workshops (MDMW’08); Beijing. 2008. p. 55-62.
9. Shyamala K, Sunitha Rani T. An analysis on efficient resource allocation mechanisms in cloud computing. Indian Journal of Science and Technology. 2015; 8(9):814-21.
10. Yiu M, Jensen C, Huang X, Lu H. Space twist: Managing the trade-offs among location privacy, query performance and query accuracy in mobile services. IEEE 24th International Conference on Data Engineering; 2008. p. 366-75.
11. Khoshgozaran A, Shahabi C. A taxonomy of approaches to preserve location privacy in location-based services. Int J Computational Science and Engineering. 2010; 5(2):86-96.
12. Goyal V, Pandey O, Sahai A, Waters B. Attribute based encryption for fine-grained access control of encrypted data. Proceedings of the 13th ACM conference on Computer and Communications Security; 2006. p. 89–98.
13. Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute based encryption. IEEE Symposium on Security and Privacy; 2007. p. 321–34.
14. Stallings W. Network Security Essentials: Applications and Standards. 4th ed. South Asia: Pearson Education; 2012.
15. Rajarajeswari S, Srinivas T. Empirical analysis of location based service approaches to preserve privacy: A cloud computing approach. International Journal of Applied Engineering Research. 2015; 10(9):6803-6.
16. Li XY, Jung T. Search me if you can: Privacy-preserving location query service; Proceedings IEEE INFOCOM; Turin. 2013. p. 2760-8.