ISSN:0975-9646

K.B.Anusha et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (6) , 2015, 4997-5003

# A Privacy Policy for Continuous Query Processing through Location Based Services

K.B.Anusha[1], M.Sweta Harini[2]
[1,2]*Assistant professor, CSE, AITAM, Tekkali, Andhra Pradesh, India.*

***Abstract:*** **With recent technological advancements in mobile devices, such as smart phones and tablets, Location-Based Services (LBSs) have surfaced as prominent applications in mobile networks. An important challenge in the wide deployment of location-based services (LBSs) is the privacy-aware management of location information, providing safeguards for location privacy of mobile clients against vulnerabilities for abuse. This paper describes a scalable architecture for protecting the location privacy from various privacy threats resulting from uncontrolled usage of LBSs. This architecture includes the development of a personalized location anonymization model and a suite of location perturbation algorithms. In particular, our algorithm makes use of a variable-sized cloaking region that increases the location privacy of the user at the cost of additional computation, but maintains the same traffic cost. Our proposal does not require the use of a trusted third-party component, and ensures that we find a good compromise between user privacy and computational efficiency. we propose a user-defined privacy grid system called dynamic grid system (DGS); the first holistic system that fulfils four essential requirements for privacy-preserving snapshot and continuous LBS. Our experiments show that the personalized location k-anonymity model, together with our location perturbation engine, can achieve high resilience to location privacy threats without introducing any significant performance penalty. Experimental results show that our DGS is more efficient than the state-of-the-art privacy-preserving technique for continuous LBS.**

***Keyword*****: - location privacy, Dynamic grid systems, k-anonymity, security etc….**

## I. INTRODUCTION

Location-based services provide convenient information access for mobile users who can issue location-based snapshot or continuous queries to a database server at anytime and anywhere. Examples of snapshot queries include "where my nearest gas station is" and "what are the restaurants within one mile of my location", while examples of continuous queries include "continuously report my nearest police car" and "continuously report the taxis within one mile of my car". Although location-based services promise safety and convenience, they threaten the security and privacy of their customers. The use of LBS, however, can reveal much more about a person to potentially untrustworthy service providers than many people would be willing to disclose. By tracking the requests of a person it is possible to build a movement profile which can reveal information about a user's work (office location), medical records (visit to specialist clinics), political views (attending political events), etc.

To tackle the privacy threats in location-based services, several spatial cloaking algorithms have been proposed for preserving user location privacy The key idea of spatial cloaking algorithms is to blur the exact user location information into a spatial region that satisfies certain privacy requirements. Privacy requirements can be represented in terms of k-anonymity (i.e., a user location is indistinguishable among k users) and/or minimum spatial area (i.e., a user location is blurred into a region with a minimum size threshold). On the other hand, our proposed technique hides query contents from the LBS, and leaves no useful clues for determining the user's current location. When a typical mobile phone accesses a third-party LBS provider through a wireless 3G data connection, we assume that it reveals only its identity and the query itself to the provider. Unavoidably, a mobile communications carrier is always aware of the user's location based on the cell towers in contact, and so it must not collude with the LBS provider. Our assumption relies on the LBS provider not being integrated into the carrier's infrastructure, such as a traffic reporting service using cell tower data that discovers a user's location passively.

Our assumption is valid for the vast majority of LBS applications, which are unaffiliated with the carrier; these include search portals, social applications, travel guides, and many other types. When communicating with such an application, the mobile user's IP address is of no help in determining the user's physical location, as it is dynamically assigned independent of location. Only a central gateway that is administered by the telecommunications carrier will be identified. We assume that no other information will be gleaned by the LBS provider. In the case where a mobile user utilizes Wi-Fi instead, the user will be assigned an address that points to the nearby access point, however, and may need to employ other techniques, such as Tor, to mask the address.

In this paper, we propose a user-defined privacy grid system called *dynamic grid system* (DGS) to provide privacy-preserving *snapshot* and *continuous* LBS. The main idea is to place a *semi-trusted* third party, termed *query server* (QS), between the user and the service provider (SP). QS only needs to be semi-trusted because it will not collect/store or even have access to any user location information. *Semi-trusted* in this context means that while QS will try to determine the location of a user, it still correctly carries out the simple matching operations required in the protocol, i.e., it does not modify or drop messages or create new messages. An *untrusted* QS would arbitrarily modify and drop messages as well as inject fake

messages, which is why our system depends on a *semi-trusted* QS.

The main idea of our DGS**.** In DGS, a querying user first determines a *query area*, where the user is comfortable to reveal the fact that she is somewhere within this query area. The query area is divided into equal-sized grid cells based on the dynamic grid structure specified by the user. Then, the user encrypts a query that includes the information of the query area and the dynamic grid structure, and encrypts the identity of each grid cell intersecting the required search area of the spatial query to produce a set of *encrypted identifiers*. Next, the user sends a request including (1) the encrypted query and (2) the encrypted identifiers to QS, which is a semi-trusted party located between the user and SP. QS stores the encrypted identifiers and forwards the encrypted query to SP specified by the user. SP decrypts the query and selects the POIs within the query area from its database. For each selected POI, SP encrypts its information, using the dynamic grid structure specified by the user to find a grid cell covering the POI, and encrypts the cell identity to produce the encrypted identifier for that POI. The encrypted POIs with their corresponding encrypted identifiers are returned to QS. QS stores the set of encrypted POIs and only returns to the user a subset of encrypted POIs whose corresponding identifiers match any one of the encrypted identifiers initially sent by the user. After the user receives the encrypted POIs, she decrypts them to get their exact locations and computes a query answer. Because the user is continuously roaming she might need information about POIs located in other grid cells (within the query area) that have not been requested from QS before. The user therefore simply sends the encrypted identifiers of the required grid cells to QS. Since QS previously stored the POIs within the query area together with their encrypted identifiers, it does not need to enlist SP for help. QS simply returns the required POIs whose encrypted identifiers match any one of the newly required encrypted identifiers to the user. After the user received the encrypted POIs from QS, she can evaluate the query locally. When the user unregisters a query with QS, QS removes the stored encrypted POIs and their encrypted identifiers. In addition, when the required search area of a query intersects the space outside the current query area, the user unregisters the query with QS and re-issues a new query with a new query area.

Contributions: Our DGS has the following key features: (1) *No TTP***.** Our DGS only requires a *semi-trusted* query server (QS) (i.e., trusted to correctly run the protocol) located between users and service providers. (2) *Secure location privacy.* DGS ensures that QS and other users are unable to infer any information about a querying user's location, and the service provider SP can only deduce that the user is somewhere within the user-specified query area, as long as QS and SP do not collude. (3) *Low communication overhead.* The communication cost of DGS for the user does not depend on the user-specified query area size. It only depends on the number of POIs in the grid cells overlapping with a query's required search area. (4) *Extensibility to various spatial queries.* DGS is applicable to various types of spatial queries without changing the algorithms carried out by QS or SP if their answers can be abstracted into spatial regions, e.g., reverse-NN queries and density queries.

## II. RELATED WORK

There are many researchers concentrating on the how to obtain the privacy and accuracy in LBSs One of the researchers was Dewri, who has a long history in the field of privacy in location-based services. He has various publications relating to achieving the privacy in LBSs His last paper [1] proposed a user-controlled privacy experience "a user-centric location based service architecture", where the user determines the desired level of privacy based on his accuracy requirements. A provider "*privacy-supportive LBS*" provides supplemental information to the user for making "*informed*" privacy decisions. The system will inform the user of the accuracy (or lack thereof) based on the privacy specifications input into the system, depending on "a service-similarity profile" which the user gets. If the user is satisfied with the result set (even if it has errors or the privacy is under the required level), they can choose to proceed with the query. If they are not satisfied, they can change the privacy level into the balance of accuracy/privacy that is acceptable to them. The main purpose of previous papers is to understand (LBS) technology and identified the key components behind the service. Some papers present a concise survey of location based services, the technologies deployed to track the mobile user's location, the accuracy and reliability associate with such measurements, and the network infrastructure elements deployed by the wireless network operators to enable these kinds of services. Other papers define the user requirements in terms of mobile device features and LBS applications.

In addition to the general idea of the LBS, the researchers discussed the impact on consumer, and utility computing offer attractive financial and technological advantages. As an example, Zhang and Mao studied the effects of three individual level factors; consumption values, privacy concerns, and subjective norms on consumers' intention to adopt location-based services on their mobile phones and to spread positive word-of-mouth (WOM) about LBS. Such knowledge helps business create effective communications to attract more potential adopters. In light of the current findings, marketing communications need to heighten perceived consumption values about using LBS.

All these scientific papers give the attracted people a general idea about LBSs, and how this service was important. Researchers have long been aware of the potential privacy risks associated with LBSs, because they know while the user used one of these application services to retrieve the accuracy information, this new functionality comes with significantly increased risks to personal privacy. They have proposed a number of promising schemes that can help users protect their privacy. Some of these papers present an overview of different protection goals and fundamental location privacy approaches, as well as a classification of different types of attacks according to the applied attacker knowledge. They clarified different protection goals and fundamental location privacy

approaches, as well as a classification of different types of attacks according to the applied attacker knowledge. The aim of these papers are to revisit the location privacy problem with the objective of providing significantly more stringent privacy guarantees.

There are several works achieving privacy-preserving location queries while using lots of different techniques for securing the location privacy being highlighted. Privacy-preserving location has three main concepts; the concept of dummy node, the concept of cloaking-region, and the concept of encryption location. However, many of these researches have a problem where the quality of the LBS and Quality of Service (QoS) decreased when anonymity is improved. The next sections will cover researches on these concepts.

## III. SYSTEM ARCHITECTURE

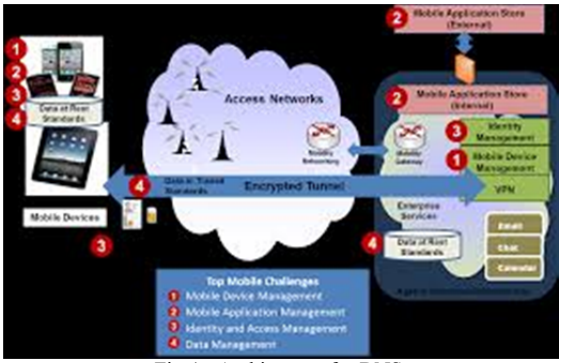


Fig-1: Architecture for DNS

Fig. depicts the system architecture of our dynamic grid system (DGS) designed to provide privacy-preserving continuous LBS for mobile users. Our system consists of three main entities, *service providers*, *query servers* and *mobile users*. We will describe the main entities and their interactions, and then present the two spatial queries, i.e., range and k-nearest-neighbour (NN) queries, supported by our system.

Service providers (SP): Our system supports any number of independent service providers. Each SP is a spatial database management system that stores the location information of a particular type of *static* POIs, e.g., restaurants or hotels, or the store location information of a particular company, e.g., Starbucks or McDonald's. The spatial database uses an existing spatial index (e.g., R-tree or grid structure) to index POIs and answer range queries (i.e., retrieve the POIs located in a certain area). As depicted in Fig. 1, SP does not communicate with mobile users directly, but it provides services for them indirectly through the query server (QS). Mobile users: Each mobile user is equipped with a GPS-enabled device that determines the user's location in the form (xu, yu). The user can obtain snapshot or continuous LBS from our system by issuing a spatial query to a particular SP through QS. Our system helps the user select a query area for the spatial query, such that the user is willing to reveal to SP the fact that the user is located in the given area.

A grid structure is created and is embedded inside an encrypted query that is forwarded to SP, it will not reveal any information about the query area to QS itself. In addition, the communication cost for the user in DGS does not depend on the query area size. This is one of the key features that distinguish DGS from the existing techniques based on the fully-trusted third party model. When specifying the query area for a query, the user will typically consider several factors. (1) The user specifies a minimum privacy level, e.g., city level. For a snapshot spatial query, the query area would be the minimum bounding rectangle of the city in which the user is located. If better privacy is required, the user can choose the state level as the minimum privacy level (or even larger, if desired). The size of the query area has no performance implications whatsoever on the user, and a user can freely choose the query area to suit her own privacy requirements. For continuous spatial queries, the user again first chooses a query area representing the minimum privacy level required, but also takes into account possible movement within the time period t.

Our contribution is based on three fundamental points; **First:** while Dewri's matrix measurements was 320 × 320 grid covered 32 kms, where each cell reflects to 100 × 100 m area with 124.5 KB data transferred. This measurement of each cell will not achieve the accuracy that the user is looking for, as well as providing the user with unnecessary needed information. Figure 3 demonstrated the major idea about the previous restriction. Suppose the user was in location (x, y) and his inquires was about some restaurant or coffee, Dewri's system will provide him a matrix about all the red spots, which is far from his interest. In fact, what he need is just an information about the nearest neighbour from his location. As a result, we zoom this area to attain the goal of accuracy while maintaining the same quantity of transmitted data, that is described in the rectangle shape in the same figure. The new similarity matrix utilized the main concept of Dewri's matrix 320 × 320 grid - where we will still in (124.5 KB) transferred data -, but each cell assimilates to 10 × 10 m area. This new cell will achieve the accuracy and efficiency results for user

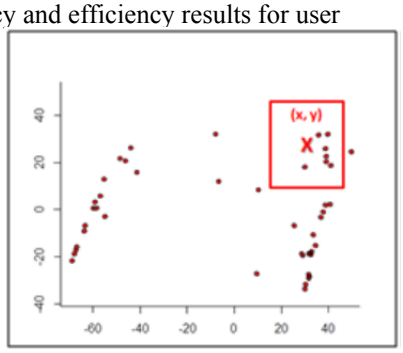


Fig-2: The New Region that the Similarity Matrix Should Covered

When the user look for a specific location around his area, the application will provide the user with the necessary information he needs. With the advent similarity matrix, the user location will be exposed, thus losing his privacy. So, the important question comes here, how we will preserve the user location? This question guided us to our **second** contribution. The answer to this question will depend on hiding the user location by making the original location anonymous (x, y) to produce a new (x`, y`). The relationship between the coordinates exemplified in

Puttaswam, where he defined that the user transforms his real-world coordinate (x, y) to a virtual coordinate (x`, y`) using his secret rotation angle and secret shift (b). Fig. (2) Illustrates the idea of anonymous location for the user.

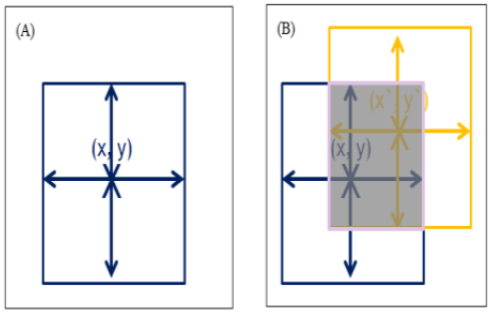


Fig-3: Anonymous User Location. (A) Clarified the Main Idea about Changing User Location. (B) The Intersection Area between Two Regions.

## Adversarial Models

We now discuss adversarial models regarding QS and SP, and then present the formal security proof of our DGS in Section 4. A malicious QS or SP will try to break a user's privacy by working with the data available to them within the described protocol. We do not consider QS or SP with access to external information not directly related to the protocol. User Anonymity As described above, both QS and SP will try to de-anonymize a user by using the information contained in the protocol (although they still faithfully follow the protocol itself). While QS does not have any information about a user that would allow it to narrow down the list of users that would fit a specific query, SP has access to the plaintext query of a user. This query, however, only contains the query region and the grid parameters, and with the information available, QS can therefore do no better than establish that the user is somewhere within the query region.

One other concern regarding the de-anonymization of users is that if for example the services of SP are paid services, then SP might for example be able to link a query with a billing record and at least establish the presence of a user in a query area. While in this paper we consider it acceptable that a user can be located to be within a query region by QS (after all, the user can freely choose the query area and hence choose it such that her personal privacy requirements are met), there is other research which would allow to prevent the linking of a query area to a specific user through billing records, for example the work by Yau and An. So even if the SP requires the authentication of users to provide a (paid) service, the service can be provided while protecting the anonymity of the user. However, no matter in which way the SP provides the service, the privacy guarantees will always be better than TTP, as a TTP always knows the exact location of the users, while in our system neither QS nor SP know the exact location of a user. Regarding paid services and QS, in such a case QS does not have any information to narrow down the geographic location of a user, even if it is being used as a paid service and can link queries to billing records. Regarding the de-anonymization of users, we also note that the type of POI in a query sent to SP or the density of POIs per cell in the query area, do not provide QS with any meaningful information that could be used to reduce the anonymity set of a user. Specifically, there is no correlation between the density of POIs in a cell and the actual location of a user, as the user launches a query without a-prior knowledge of the density of POIs, and hence the density of POIs in a cell cannot be used to make deductions about the possible location of a user in the query area. A natural choice for the role of QS is the network service provider of a user. Even though the network service provider can typically locate a user down to an individual cellular network cell already, taking on the role of QS does not provide it with any additional information about the user, such as the actual query area. There is also no requirement for the queries of a user to correspond to the user's actual location, and the network service provider does not have any information available that would allow it to infer either case. To summarize, a network service provider already knows the location of its users, and serving as a QS does not provide it with any additional information about its users. Alternatively, QS services could also be provided by volunteers (e.g., like many of the nodes in the Tor network), by ad-supported services, or even by services that charge a modest fee. *2.1.2 Other Attacks* In this subsection we discuss a few other attacks and explain how they relate to our proposed system.

IP localization: One possible attack involves QS trying to determine the position of a user through IP localization (i.e., using a database which can map IP addresses to locations). Because of how mobile phone networks are setup (considering that our system is aimed at mobile users using mobile phone networks), however, mobile phones cannot be located with useful accuracy, as shown by Balakrishnan et al. [20]. Even so, if IP localization is a concern, solutions at the network level can hide the originating IP, for example by using an anonymizing software such as Tor.

Timing attacks: Another set of attacks might use timing information if QS can observe the traffic close to the originating user. However, if QS can observe such traffic, the location privacy of the user is very likely already compromised, even without timing attacks. Furthermore, we consider this to be out of the scope of our work.

Query server as client: QS might try to also act as a client in an attempt to gain some information which could help to localize a user. QS has no information to launch such an attack, however, not even an approximate location of the user. Also, the number of POIs returned to a client does not allow QS to make any inferences, because it knows neither the query area nor the grid parameters. A large number of POIs could either mean a dense region or a large query area, depending on the grid parameters, which are unknown to QS.

Network traffic fingerprinting: An attack as described by Bissiaset al. which makes inferences based on the statistics of encrypted network connections is not applicable to our system. The attack as described in the paper is equivalent to determining which QS a user is using. This information does not need to be secret and communication with QS is highly uniform across different query servers (unlike

website traffic), very likely making them for all practical purposes indistinguishable.

Commuter problem: Another attack that can identify the home and the office location of a user through location traces is described by Golle et al. This attack is not applicable to our system, because no plaintext locations are ever transmitted in our system and no inferences can be made. We exclude side-channel attacks in general from the security analysis as being out of the scope of this paper. Many of the side-channel attacks mentioned above are fundamental to network communications, and as such neither limited to nor a consequence of the design of our proposed protocol.

## IV. EXPERIMENTAL RESULTS

Similar to continuous range queries, the privacy-preserving query processing for continuous k-NN queries has two main phases. The first phase finds an initial (or snapshot) answer, while the second phase maintains the correct answer when the user moves by using incremental updates . However, unlike range queries, the required search area of a k-NN query is unknown to a user until the user finds at least k POIs to compute a required search area, i.e., a circular area centred at the user's location with a radius from the user to the $K^{th}$-nearest POI. Thus, the privacy-preserving query processing protocol of k-NN queries is slightly different.

K-Nearest-Neighbour Query Processing: A continuous K-NN query is defined as keeping track of the K-nearest POIs to a user's current location (xu, yu) for a certain time period, as presented in Section 2. In general, the privacy preserving K-NN query processing has six major steps to find an initial (or snapshot) query answer. Fig. 4 depicts a running example of the privacy-preserving query processing of a K-NN query, where k = 3.

Step 1.Dynamic grid structure (by the user): This step is the same as the dynamic grid structure step (Step 1) in the range query processing phase (Section 3.1.1). It takes a user-specified query area with a left-bottom vertex (xb, yb) and a right-top vertex (xt, yt) and divides the query area into m×m equal-sized cells, as illustrated in Fig. 4a (m = 6).

Step 2.Request generation (by the user): The required search area of the k-NN query is initially unknown to the user. The user first finds at least k POIs to compute the required search area as a circular area centred at the user's location with a radius of a distance from the user to the k-th nearest known POI.The user therefore first attempts to get the nearby POIs from a specific SP. In this step, the user requests the POIs in the cell containing the user and its neighbouring cells from SP. Given the user's current location (xu, yu) and a query area specified by the user in Step 1, she wants to get the POIs within a set of grid cells Sc that includes the cell containing herself, i.e., (cu, ru) = _j xu−xb (xt−xb)/mk , j yu−yb (yt−yb)/mk_, and its at most eight neighbouring cells (cu −1, ru−1), (cu, ru−1), (cu+1, ru−1), (cu − 1, ru), (cu + 1, ru), (cu − 1, ru + 1), (cu, ru + 1), and (cu + 1, ru + 1). For each cell i in Sc, the user generates an encrypted identifier Ci using Equations 3 and 4, as in the request generation step (Step 2) in the range query processing phase. The user also creates a query to be sent to SP based on Equation 2. Finally, the user sends a request, which includes the identity of SP, the query, and the set of encrypted identifiers (in random order) Se, as given in Equation 5, to QS. In the running example (Fig. 4a), the user located in the grid cell (3, 3) and therefore requests the POIs in the cells (3, 3) and its neighbouring grid cells, i.e., (2, 2), (3, 2), (4, 2), (2, 3), (4, 3), (2, 4), (3, 4), and (4, 4), (represented by shaded cells) from SP through QS.

Step 3.Request processing (by QS)**:** This step is identical to Step 3 for range queries in the query processing phase (Section 3.1.1).

Step 4.Query processing (by SP): This step is identical to Step 4 for range queries in the query processing phase (Section 3.1.1). Thanks to this query abstraction feature, our DGS can be easily extended to support other continuous spatial query types, e.g., reverse NN queries and density queries.

Step 5.Required search area (by the user and QS**):** This step is similar to the encrypted identifier matching step (Step 5) for range queries in the query processing phase (Section 3.1.1), with the difference that this step may involve several rounds of interaction between the user and QS. QS Matches the encrypted identifiers of the encrypted POIs returned by SP with the encrypted identifiers in Se sent by the user in Step 2, and sends the matching encrypted POIs to the user. If at least k encrypted POIs are returned to the user, she can decrypt them to compute a required search area for the K-NN query in the form of a circle centred at the user's location with a radius of the distance between the user and the $K^{th}$-nearest POI. On the other hand, if less than K POIs are returned, the user starts the next iteration by requesting the grid cells from QS one hop further away from the position of the user, i.e., the neighbouring cells of the grid cells that have already been requested by the user. This incremental search process is repeated (i.e., requesting more cells moving steadily outward from the user's position) until the user has obtained at least k POIs from QS. After the user determines the required search area, there are two possibilities: 1) the user has already requested all the cells which intersect the required search area. In this case, the user proceeds to the next step. 2) The required search area intersects some cells which have not yet been requested from QS. The (at least) k POIs found so far may in that case not be an exact answer, and the required search area but have not been requested yet (in Fig. 4c these would be the shaded cells outside the bold rectangle). After receiving all encrypted POIs in the newly requested cells from QS, the user proceeds to the next step. In the running example, Fig. 4b depicts that the grid cells initially requested by the user (within the bold rectangle) contains less than three POIs. The user therefore requests the neighbouring grid cells (the grid cells adjacent to the bold rectangle) from QS. This will result in discovering three POIs in total, i.e., p1, p2, and p3. The user then computes the required search area represented by a circle (Fig. 4c). As the required search area intersects eight cells which the user has not yet requested from QS (i.e., they are outside the bold rectangle), the user will launch another request for

the grid cells (0, 1), (0, 2), (0, 3), (0, 4), (1, 0), (2, 0), (3, 0), and (4, 0).

Step 6. Answer refinement (by the user). Having received all the POIs within all the grid cells intersecting the required search area, the user can decrypt them to get their exact locations, as in the answer computation step (Step 6) for range queries in the query processing phase (Section 3.1.1), and determine the exact answer

by selecting the k nearest POIs. The previous steps ensure that these k POIs are indeed the closest ones. In the running example (Fig. 4d), the user can find the exact answer for the 3-NN query, which includes three POIs p1, p2, and p4.

*3.2.2 Incremental K-NN Query Answer Maintenance* After the user computes an initial (or snapshot) k-NN query answer, the incremental answer update phase allows to maintain the answer as the user moves around. Similar to range queries, the incremental answer maintenance phase has four steps. The first two steps are the same as the cache region step and the incremental request generation step as in Section 3.1.2. In the third step (i.e., request processing) performed by the QS, since QS has already cached the encrypted POIs, together with their corresponding encrypted identifiers calculated by SP in Step 4 of the query processing phase, it does not need to contact SP. It can simply forward the encrypted POIs matching one of the encrypted identifiers in Se to the user. In the last step (i.e., answer refinement) performed by the user, she decrypts the received POIs and sorts the ones located within the required search area according to their distance to the user in ascending order. The k-nearest POIs to the user constitute the new query answer.

We summarize major findings from our experiments and the insights obtained from the experimental results in four points:

1. NBR-k out performs local-k in both success rate and relative anonymity level metrics without incurring extra processing overhead. This is due to its ability to anonymize larger groups of messages together at once.
2. The deferred search, a technique that aims at decreasing the number of clique searches performed in an effort to increase runtime performance, turns out to be inferior to the immediate search. This is because, for smaller k values, the index search and update cost is dominant over the clique search cost and the deferred search increases the size of the index due to batching more messages before performing the clique searches.
3. The progressive search improves the runtime performance of anonymization, especially when constraint boxes and k values are large, without any side effects on other evaluation metrics. This nature of the progressive search is due to its proximity-aware nature: The close-by messages that are more likely to be included in the result of the clique search are considered first with the progressive search.
4. The Clique Cloak algorithms have the nice property that, for most of the anonymized messages, the cloaking box generated is much smaller than the constraint box of the received message specified by the tolerance values, resulting in higher relative spatial and temporal resolutions. In conclusion, the configuration of [nbr-k, immediate, progressive] is superior to other alternatives.

## V. FUTURE WORK

Our personalized k-anonymity model requires mobile clients to specify their desired location anonymity level and their spatial/temporal tolerance constraints. It is possible that the level of privacy and the QoS can be in conflict in a user's specification. When such conflicts occur, the success rate of anonymization will be low for this user's messages. In practice, such conflicts should be checked to determine the need for fine-tuning in the privacy level or QoS. The trade-off between the QoS defined by the spatial/ temporal tolerance constraints and the level of privacy protection defined by the anonymity level k should be adjusted such that the success rate of anonymization is kept close to 1. In this paper, we developed a location anonymization framework and associated system-level facilitates for fine-tuning of the QoS versus privacy protection trade-off. Due to the space constraint, we did not discuss the application-dependent management of user involved adjustment of this trade-off. We believe that these issues merit an independent study.

### 5.1 Optimality of the Clique Cloak Algorithms:

It is important to note that the Clique Cloak algorithms that we introduced in this paper are heuristic in nature. Although we do not know the best success rates that can be achieved for various distributions of anonymity constraints, we experimentally showed that, for practical scenarios in the worst case, our algorithms drop only 10 percent of the messages due to non optimality. Furthermore, since it is extremely hard to accurately predict future patterns of messages, it is difficult to build an online optimal algorithm. These two observations lead us to the conclusion that our algorithms will be highly effective in practice. However, it is an open problem to study advanced algorithms that have better optimality and runtime performance.

### 5.2 Pseudonymous and Non anonymous LBSs:

In this paper, we assumed that the LBSs are anonymous; that is, the true identities of mobile clients are not required in the services provided. Services that require the knowledge of user identities or pseudonyms (non anonymous and pseudonymous LBSs) will make the tracking of successive messages from the same users trivial at the LBS side. We believe that the pseudonymous LBSs can benefit from our solution with some modifications. For instance, one complication may arise when successive location-identity bindings take place and the set of k messages from the two adjacent bindings share only one pseudonym, which can easily lead to a trajectory-identity binding. These types of vulnerabilities can be prevented or mitigated by setting proper time intervals for changing the pseudonyms associated with mobile clients, without violating the service requirements of the LBSs. Nevertheless, further research is needed for devising effective techniques for performing privacy-preserving

pseudonym updates. In the case of non anonymous LBSs, we believe that the location privacy protection will need to be guaranteed through policy-based solutions managed by LBS providers. Policy-based solutions require mobile clients to completely trust the LBS providers in order to use the services provided.

After going through the surveying, it can be gathered that there is a huge scope of application development in mobile domain. Following the same notion, we can also develop application that can tackle following issues:

1) Location positioning technologies
2) Query processing
3) Cache management

The LBS application can help user to find hospitals, school, gas filling station or any other facility of interest indicated by user within certain range. Just like a GPS device its location will also be updated as soon as user changes his/her position.

## VI. CONCLUSION

We proposed a personalized k-anonymity model for providing location privacy. Locations based services promise a very bright future considering all the key aspects of technologies required to operate the LBS available in the market. Moreover, the number of people that it can reach is far from expectation due to the number of mobile users around the world. In this paper, we proposed a supplement architecture which successfully solves the privacy issues in existing LBS applications and provides a new system, We developed an efficient message perturbation engine to implement this model. Our message perturbation engine can effectively anonymize messages sent by the mobile clients in accordance with location k-anonymity while satisfying the privacy and QoS requirements of the users. Several variations of the spatio-temporal cloaking algorithms, collectively called the Clique Cloak algorithms, are proposed as the core algorithms of the perturbation engine. Our work continues along a number of directions, including the investigation of more optimal algorithms under the proposed framework, the study of QoS characteristics of real-world LBS applications, and how QoS requirements impact the maximum achievable anonymity level with reasonable success rate. The system achieved better performance by not threatening the accuracy of the system without the requirements of providing results such as sparse level. Allowing the user to have complete flexible control over their privacy and their system, took the matrix to a whole new better bandwidth level.

## REFERENCES


[1]. Roman Schlegel, Chi-Yin Chow, Qiong Huang, and Duncan S.Wong."User-Defined Privacy Grid System for Continuous Location-Based Services" DOI 10.1109/TMC.2015.2388488, IEEE Transactions on Mobile Computing.

[2]. Reemah M. Alhebshi, Jonathan Cazalas."Improving the Similarity for Privacy in Location Based Service"*International Journal of Computer and Information Technology (ISSN: 2279 – 0764) Volume 03 – Issue 06, November 2014.*

[3]. Femi Olumofin, Piotr K. Tysowski, Ian Goldberg, and Urs Hengartner." Achieving Efficient Query Privacy for Location Based Services" www.cypherpunks.ca/~iang/pubs/lbspir-pets.pdf.

[4]. Chi-Yin Chowand Mohamed F.Mokbel. "Enabling Private Continuous Queries For RevealedUserLocations"www.cs.ucsb.edu/~ravenben/ classes/595n-s07/papers/sstd07.pdf

[5]. Bugra Gedik and Ling Liu."Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms" IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 7, NO. 1, JANUARY 2008.

[6]. Virrantaus,K, Markkula.J, Garmash.A., Terziyan.V, Veijalainen.J, Katanosov.A and Tirri.H. "Developing GIS supported location-based services" in Web Information Systems Engineering (2001), IEEE, pp. 66_75.

[7]. Consortium, O. G. Open location services 1.1, 2005.

[8]. D'Roza, T., and Bilchev, G. An overview of location-based services. BT Technology Journal 21, 1 (2003), 20_27

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

[10]. P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, "Preventing location-based identity inference in anonymous spatial queries," *IEEE TKDE*, vol. 19, no. 12, pp. 1719–1733, 2007.

[11]. M. F. Mokbel, C.-Y. Chow, and W. G. Aref, "The new casper: Query processing for location services without compromising privacy," in *VLDB*, 2006.

[12]. T. Xu and Y. Cai, "Location anonymity in continuous location-based services," in *ACM GIS*, 2007 [8].Schwinger, W., Grin, C., Prll1, B., and Retschitzegger, W. A light-weight framework for location-based services. In Lecture Notes in Computer Science (Berlin, 2005), Springer, pp. 206_210 [5].Zeimpekis, V., Giaglis, G,and Lekakos, G. A taxonomy of indoor and outdoor positioning techniques for mobile location services. SIGecom Exch. 3, 4 (2003), 19_27

[13]. H. S.-M. Ali Khoshgozaran and C. Shahabi. SPIRAL, a scalable private information retrieval approach to location privacy. In Proceedings of the 2nd International Workshop on Privacy-Aware Location-based Mobile Services (PALMS), 2008.

[14]. B.Bamba, L.Liu, P.Pesti, and T.Wang. Supporting anonymous location queries in mobile environments with privacygrid. In Proceeding of the 17th international conference on World Wide Web, pages 237–246, New York, NY, USA, 2008.

[15]. A. Beimel and Y. Stahl. Robust information-theoretic private information retrieval. J. Cryptol., 20(3):295–321, 2007.

[16]. C. Bettini, S. Jajodia, P. Samarati, and X. S. Wang, editors. Proceedings of the 1st International Workshop on Privacy in Location-Based Applications, Malaga, Spain, October 9, 2008, volume 397 of CEUR Workshop Proceedings, 2008.


## AUTHORS


1. K.B.Anusha is working as Asst. Professor in the Department of Computer Science & Engineering of AITAM Engineering College, Tekkali (India). She has 1 year of experience in the field of Academics and is actively involved in research & development activities. She obtained his Master of Technology degree in (Computer Science). She has worked as lecturer in Aditya Institute of technology and Management , Tekkali.

2. M. Swetha Harini is working as Asst. Professor in the Department of Computer Science & Engineering of AITAM Engineering College, Tekkali(India). She has 1 year of experience in the field of Academics and is actively involved in research & development activities. She has published papers in conference and attended seminar. She has worked as a Teaching Assistant in Aditya Institute of technology and Management for 1 year.