User-Defined Privacy Grid System For Continuous Location-Based Services

[1]Prof. Sumeet Pate, [2]Miss. Chande Vaishali S, [3]Miss. Karale Rohini B, [4]Miss. Narvekar Shivali J

[1]Asst. Professor, [2,3,4]UG Student, [1,2,3,4]Computer Engg. Dept. Shivajirao S.Jondhle College of Engineering & Technology, Asangaon, Maharshatra, India.

[2]vchande7@gmail.com, [3]rohinikarale@gmail.com, [4]shivalinarvekar@gmail.com

Abstract— Location-based services (LBS) require users to continuously report their location to a potentially untrusted server to obtain services based on their location, which can lead them to privacy risks. Unfortunately, present privacy-preserving techniques for Location Based Services have numerous restrictions, such as requiring a fully-trusted third party (FTTP), providing limited privacy guarantees and incurring high communication overhead. A user-defined privacy grid system called dynamic grid system (DGS) the first holistic system that fulfills four essential requirements for privacy-preserving snapshot and continuous Location Based Services. The system only requires a semi-trusted third party, blamable for carrying out simple matching operations correctly. This semi-trusted third party does not have any data about a user's location. Safe snapshot and continuous location privacy is guaranteed under our defined adversary models. The communication cost for the user does not depend on the user's desired privacy level, it only depends on the number of significant points of interest in the vicinity of the user. Although we only focus on range and k-nearest-neighbor (KNN) queries in this work, our system can be simply extended to support other spatial queries without changing the algorithms run by the semi-trusted third party and the database server, provided the required search area of a spatial query can be abstracted into spatial regions. Experimental outcomes show that our Dynamic Grid System is more efficient than the state-of-the-art privacy-preserving technique for continuous LBS.

Keyword: Dynamic grid system, location privacy, location based services, spatio -temporal query processing, cryptography, mobile computing.

I. INTRODUCTION

In the present world of mobility and ever- Internet connectivity, a huge number of people use LBS to obtain information related to their current locations from different types of service providers. This can be made as the search for nearest points of interests (POIs) (e.g., hotels, malls) location-aware made by companies, traffic information suggested to the highway and providing direction to the user who is traveling and so on. The use of Location based services (LBS) can provide more details about a person to potentially untrustworthy service providers behind which people might obtain it. They can track the request made by the person it is possible to make a movement profile which can give data about a user's work space , health related record , political events (attending political events, conferences), etc. Nevertheless, Location based services (LBS) can be very valuable and users should be able to make use of the services given by them, without giving up their location privacy.

A number of methods have recently been suggested for preserving the location privacy of uses in Location based services (LBS) the techniques can be divided into two main categories, Fully-trusted third party (FTTP). The most popularly used privacy-preserving techniques require a trusted third party to be placed among the user and the service provider where to hide the user's location data from it. Private data retrieval/oblivious transfer : Although private data retrieval or oblivious transfer techniques do not require a third party, they obtain greater communication overhead among the user and the service provider, requiring the transmission of many details than the user actually needs only a few privacy-preserving techniques have been suggested for continuous Location based services.

Mobile Computing is a technology that allows transmission of data, voice and video through a computer or any other wireless enabled device without having to be connected to a fixed physical link. Mobile computing is the discipline for creating an information management platform, which is free from spatial and temporal constraints. The freedom from these constraints permits its users to access and process desired data from anywhere in the space.

The state of the user, static or mobile, does not affect the information management capability of the mobile platform. A user can continue to access and manipulate desired data while traveling on plane, in car, on ship, etc. Thus, the discipline creates an illusion that the desired data and sufficient processing power are available on the spot, where as in reality they may be located far away. Otherwise Mobile computing is a common term used to discuss to a variety of devices that allow people to access data and information from where ever they are.

An Location Based Services (LBS) requires five a basic components the services provider's software application, a mobile network to transmit data or information and request for service, a content provider to supply the end user.

II. LITERATURE SURVEY

A. Enabling private continuous queries for revealed user locations

Present location-based services provide specialized services to their customers based on the knowledge of their exact locations. With untrustworthy servers, location-based services may expose to numerous privacy threats ranging from worries over employers snooping on their worker's where about to fears of tracking by potential followers. While there exist numerous techniques to preserve location privacy in mobile environments, these methods are limited as they do not distinguish among location privacy (i.e., a user wants to hide her location) and query privacy (i.e., a user can reveal her location but not her query).This distinction is crucial in many applications where the locations of mobile users are publicly known. In this paper, the restriction of existing cloaking algorithms as we intend a new robust spatial cloaking technique for snapshot and continuous location-based queries that clearly distinguishes amongst location privacy and query privacy. By this difference, we achieve two main goals:
(1) Supporting private LBS to those customers with public locations, and (2) Performing spatial cloaking on-demand basis only (i.e., when issuing queries) rather than exhaustively cloaking every single location update.

Experimental outcomes display that the robust spatial cloaking algorithm is scalable and efficient while providing anonymity for large numbers of continuous queries without hiding users' locations.

B. Protecting location privacy with personalized K Anonymity: Structural design and algorithm

Continued advances in mobile networks and positioning technologies have created a strong market push for location-based applications. Examples include location-aware emergency response, location-based advertisement, and location-based entertainment. 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 user against vulnerabilities for abuse.

This paper defines a scalable structural design for protecting the location privacy from various privacy threats resulting from uncontrolled usage of Location Based Services. This structural design consist of the development of a personalized location anonymization model and a suite of location perturbation algorithms. A unique characteristic of our location privacy structural design is the use of a flexible privacy personalization framework to support location k-anonymity for a wide range of mobile users with context-sensitive privacy requirements.

C. Anonyms usage of location-based services through spatial and temporal cloaking

Advances in sensing and tracking technology enable location-based applications but they also create significant privacy risks. Anonymity can provide a high amount of privacy, save service users from dealing with service provider's privacy policies, and decreases the service provider's requirements for safeguarding -private information. However, guaranteeing anonymous usage of LBS requires that the precise location data transmitted by a client cannot be easily used to re-identify the subject. This paper presents a middleware structural design and algorithms that can be used by a centralized location agent service.

The adaptive algorithms adjust the resolution of location information along spatial or temporal dimensions to meet specified anonymity constraints based on the entities who May be using location services within a given area. Using a model based on automotive traffic counts and cartographic material, we estimate the realistically expected spatial resolution for different anonymity constraints.

D. Preventing location-based identity interference in anonymous spatial queries

The increasing development of embedding positioning capabilities (for example, Global Positioning System (GPS)) in mobile devices facilitates the widespread use of location-based services. For such applications to succeed, privacy and confidentiality are essential. Existing privacy-enhancing techniques rely on encoding to safeguard communication channels, and on pseudonyms to protect user identities. Nevertheless, the query contents may disclose the physical location of the client. In this paper, we present a framework for preventing location-based identity inference of users who issue spatial queries to LBS. We propose transformations based on the well-established K-anonymity concept to compute exact answers for range and nearest neighbor search, without enlightening the query source. Our methods optimize the whole process of anonymizing the requests and processing the transformed spatial queries. Extensive experimental studies suggest that the proposed techniques are applicable to real-life scenarios with numerous mobile users.

***E.* Supporting anonymous location queries in mobile environment with privacy grid**

This paper presents Privacy Grid - a framework for supporting anonymous location-based queries in mobile data delivery systems. The Privacy Grid framework deals three unique capabilities. First, it provides a location privacy protection preference profile (called location P3P) model, which allows mobile users to explicitly define their preferred location privacy requirements in terms of both location hiding measures (e.g., location k-anonymity and location I-diversity) and location service quality measures (e.g., maximum spatial resolution and maximum temporal resolution). Second, it provides fast and effective location cloaking algorithms for location k-anonymity and location I-diversity in a mobile environment. We develop dynamic bottom-up and top-down grid cloaking algorithm with the goal of achieving high anonymization success rate and efficiency in terms of both time complexity and maintenance cost. A hybrid approach that carefully combines the strengths of both bottom-up and top-down cloaking approaches to further decrease the average anonymization time is also developed. Privacy Grid in-corporates temporal cloaking into the location cloaking process to further increase the profit rate of location anonymization.

Table 1. Comparative Study

Paper Names	Author	Privacy Methods used	Merits	Demerits
Supporting anonymous location queries in mobile environments with PrivacyGrid	Bamba, L. Liu, P. Pesti, and T.Wang	PrivacyGrid	fast and effective location cloaking algorithms	Performance penalty.
Enabling private continuous queries for revealed user locations	C.-Y. Chow and M. F. Mokbel	Existing location-based services	provide specialized services to their customers based on the knowledge of their exact locations	Robust spatial cloaking algorithm is scalable and efficient while providing anonymity for large numbers of continuous queries without hiding user's locations.
Protecting location privacy with personalized kanonymity: Structural design and algorithms	B. Gedik and L. Liu	Continued advances in mobile networks and positioning technologies	A strong market push for location-based applications	Performance penalty.
Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking	M. Gruteser and D. Grunwald	Advances in sensing and tracking technology	Provide similar location-dependent services	Significant privacy risks
Preventing location-based identity inference in anonymous spatial queries	P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias	a framework for preventing location-based identity	growing trend of embedding positioning capabilities	Only Extensive experimental studies is applicable

III. PROPOSED SYSTEM

A user-defined privacy grid system called dynamic grid system (DGS) to provide privacy-preserving snapshot and continuous Location Based Services (LBS). The main idea is to place a semi trusted third party, termed query server (QS), among the user and the service provider (SP). Query Server only needs to be semi-trusted because it will not collect or store or even have access to any user location data. Semi-trusted in this context means that while Query Server will try to determine the location of a user, it still correctly carries out the simple matching operations required in the protocol, that is it does not change or drop messages or create new messages. An untrusted Query Server would arbitrarily change and drop messages besides insert fake messages, which is why our system depends on a semi-trusted QS. These techniques not only inherit the disadvantage of the Trusted Third Party (TTP) model, but they also have other limitation as follows

1. Inefficiency: Continuously getting bigger cloaked areas substantially increases the query processing overhead.
2. Service Termination: A user has to terminate the service when users initially assigned to her cloaked area leave the system.
3. Privacy Leakage: Since the database server receives a set of consecutive cloaked areas of a client at different timestamps the correlation among the cloaked areas would provide useful data for inferring the user's location.

The main idea of DGS

In Dynamic Grid System (DGS), a querying user first defines a query area, where the user is comfortable to reveal the fact that she is somewhere within this query area. The query area is distributed into equal-size grid cells based on the dynamic grid structure specified by the user. Then, the user encodes a query that includes the information of the query area and the dynamic grid structure, and encodes the identity of each grid cell intersecting the required search area of the spatial query to produce a set of encoded identifiers.

Next, the user sends a request including (1) the encoded query and (2) the encoded identifiers to QS, which is a semi-trusted party located amongst the client and Service Provider. QS Store's the encoded identifiers and forwards he encoded query to SP specified by the user. Service Provider decodes the query and selects the POIs within the query area from its database.

IV. MATHEMATICAL MODEL

RANGE QUERY PROCESSING:

.Dynamic grid structure
(xb, yb) = bottom-left vertex
(xt, yt) =.top-right vertex
Grid cell is identified by (c, r),
c= column index from left to right
r= row index from bottom to top
$0 \leq c, r < m$
Given the coordinates of the bottom-left vertex of a grid cell, (xc, yc).

V. SYSTEM ARCHITECTURE

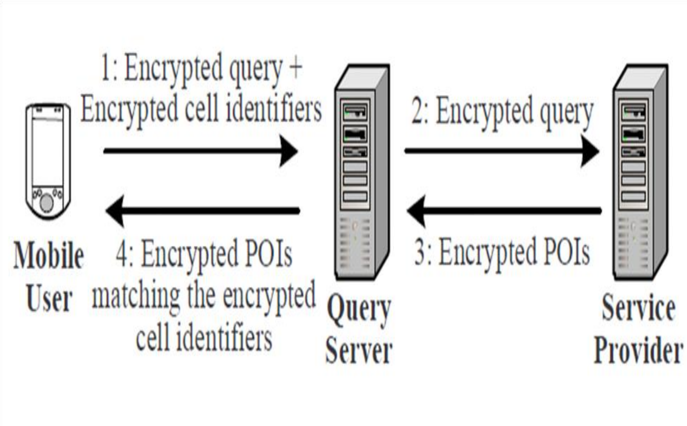


Fig 1. System Architecture

The system architecture of our dynamic grid system (DGS) designed to provide privacy-preserving continuous Location Based Services (LBS) for mobile users. Our system consists of three main entities, service providers (SP), query servers (QS) and mobile users.

Service providers (SP):
Our system supports any number of independent service providers. Each Service Provider is a spatial database management system that stores the location information of a particular type of static Point Of Interest (POIs), e.g., restaurants or hotels or the store location information of a particular company.

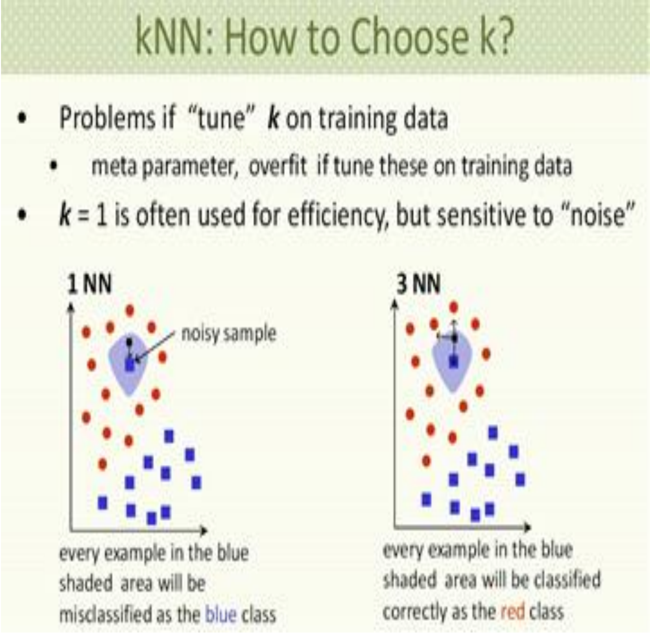
Mobile users:
Each mobile user is equipped with a Global Positioning System (GPS)-enabled device that determines the user's location in the form (xu, yu).The user can obtain snapshot or continuous Location Based Services from our system by issuing a spatial query to a particular Service Provider through Query Server.

Query servers (QS):
Query Server is a semi-trusted party placed among the mobile user and Service Provider. Similar to the most popular infrastructure in existing privacy-preserving techniques for Location Based Services, Query Server can be maintained by a telecom operator.

VI. ALGORITHM

k- nearest Neighbors algorithm for classification
k- NN
1. Determine parameter K = number of nearest neighbors
2. Calculate the distance between the query-instance and all the Training samples
3. Sort the distance and defines nearest neighbours based on the K-th minimum distance
4. Gather the category of the nearest neighbours
5. Use simple majority of the category of nearest neighbours as the prediction value of the query instance

Step1:
Calculate k
1. Before running the algorithm, decide on the value of k, that is, how Many records will have a voice in classifying the new record.
2. Then, compare the new record to the k nearest neighbors, that is, to The k records that are of minimum distance from the new record in Terms of the Euclidean distance or whichever metric the user prefers.
3. Once the k records have been chosen, then for simple an weighted voting, their distance from the new record no longer matters. It is simple one record, one vote.
Step 2:
Choosing value of k
With small k (e.g., k =1), the algorithm will simply return the target value of the nearest observation, a process that may lead



kNN: How to Choose k?

- Problems if "tune" *k* on training data
  - meta parameter, overfit if tune these on training data
- *k* = 1 is often used for efficiency, but sensitive to "noise"

every example in the blue shaded area will be misclassified as the blue class

every example in the blue shaded area will be classified correctly as the red class

Algorithm toward over fitting, tending to memorize the training data set at the expense of generalizability. On the other hand, choosing a value of k that is not too small will tend to smooth out any idiosyncratic behavior learned from the training set. However, if we take this too far and choose a value of k that is too large, locally interesting behavior will be overlooked. The data analyst needs to balance these considerations when choosing the value of k.

It is possible to allow the data itself to help resolve this problem, by following a cross-validation procedure similar to the earlier method for finding the optimal values z1,z2,...zm For axis stretching. Here we would try various values of k with different randomly selected training sets and choose the value of k that minimizes the classification or estimation error.

Step 3:

Calculate Distance

Data analysts define distance metrics to measure similarity. A distance metric or distance function is a real-valued function d , such that for any coordinates x, y, and z :

1. d(x, y)≥0, and d(x, y)=0 if and only if x=y
2. d(x, y)=d(y,x)
3. d(x,z)≤d(x,y)+d(y,z)

Property 1: assures us that distance is always non-negative, and the only way for distance to be zero is for the coordinates (e.g., in the scatter plot) to be the same.

Property 2 : indicates commutatively, so that, for example, the distance from New York Los Angeles is the same as the distance from Los Angeles to New Yo-rk.

Finally, property 3: the triangle inequality, which states that introducing a third point can never shorten the distance among two other points. The most common distance function is Euclidean distance, which represents the usual manner in which humans think of distance in the real world:

Euclidean distance treats each feature as equally important.

$d_{Euclidean}(x,y)= (\sum_i (xi - yi)2)^{1/2}$
where,
x=x1,x2,...,xm, and
y=y1,y2,...,ym represent the m attribute values of two records

Step 4:

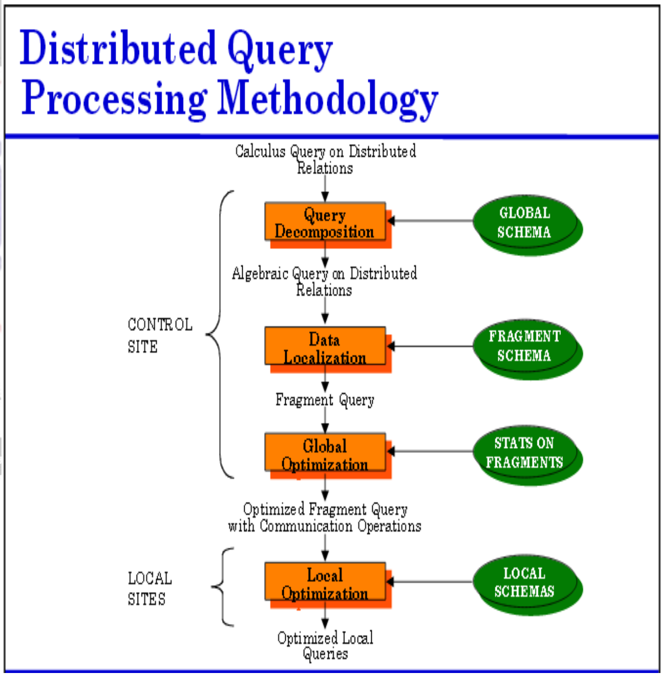
Computational Complexity Calculation

Basic kNN algorithm stores all examples

Suppose we have n examples each of dimension d then,

- O(d) to compute distance to one examples
- O(nd) to computed distances to all examples
- Plus O(nk) time to find k closest examples
- Total time: O(nk+nd)

Query Processing Algorithm



Distributed Query Processing Methodology

Fig 2: Query Processing Algorithm.

Step 1: Query Decomposition

Input: calculus query on global relation

✓ Normalization

- Manipulate query quantifiers and qualification

✓ Analysis

- Detect and reject "incorrect" queries
- Possible for only a subset relational calculus

✓ Simplification

- Eliminate redundant predicates

- ✓ Restructuring
  - Calculus query ⇒ algebraic query
  - More than one translation is possible
  - Use transformation rules

Step 2: Data Localization

Input: Algebraic query on distributed relation

- ✓ Determine which fragment are involved
- ✓ Localization program
  - Substitute for each global query its materialization program
  - Optimize

Step 3: Global Query Optimization

Input: Fragment query

- ○ Find the best(not necessarily optimal)global schedule
- ✓ Minimize a cost function
- ✓ Distributed join processing
  - Bushy vs linear trees
  - Which relation to ship where?
  - Ship whole vs ship-as-needed
- ✓ Decide on the use of semijoins
  - Semijoins saved on communication at the expense of more local processing
- ✓ Join methods
  - Nested loops vs ordered joins(Merge join or
    Hash join)

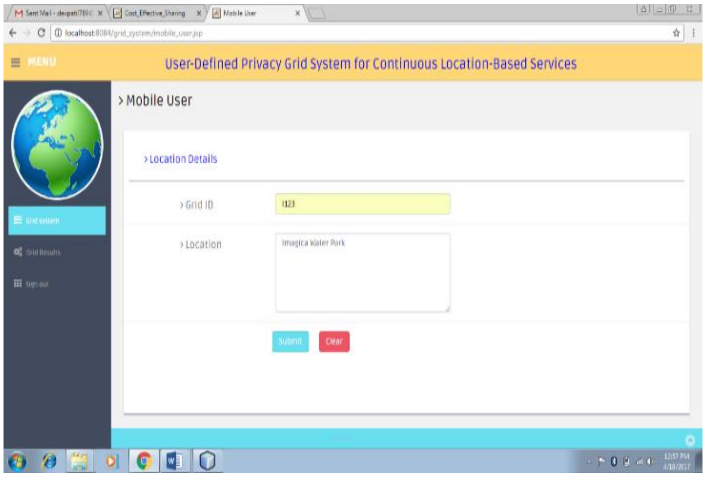
Step 4: Local optimization

Input: Best global execution schedule

- ✓ Select the best access path
  - Use the centralized optimization technique

VII. EXPECTED OUTPUT

The user will enter the grid id and location and then submit. To the query server Location which we want to see that has been grid.



localhost:8084/grid_system/mobile_user.jsp

User-Defined Privacy Grid System for Continuous Location-Based Services

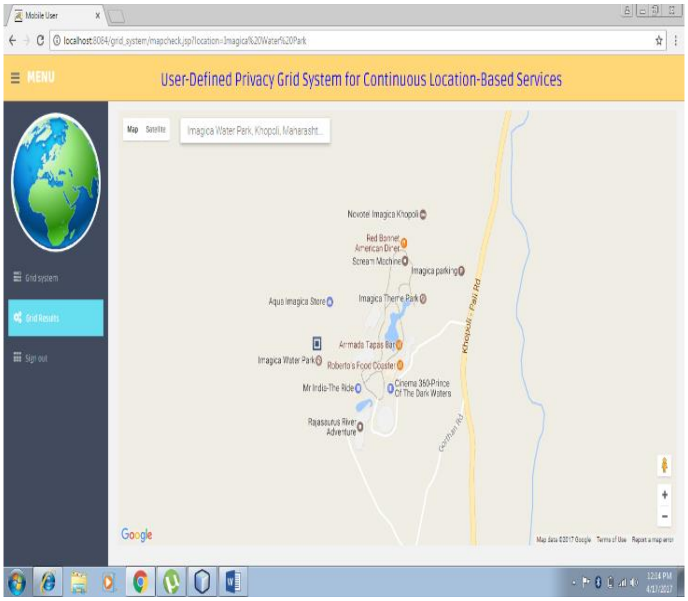
› Mobile User

› Location Details

› Grid ID: 123

› Location: Imagica Water Park

Fig 3: Input



localhost:8084/grid_system/mapcheck.jsp?location=Imagica%20Water%20Park

User-Defined Privacy Grid System for Continuous Location-Based Services

Fig 4: Output

VIII. CONCLUSION

We have tried to implemented "Roman Schlegel, Chi-Yin Chow, Qiong Huang and Duncan S. Wong","User defined Privacy Grid System For Continuous Location Based Services", IEEE TRANSACTIONS ON MOBILE COMPUTING, 2013 and according to the implementation the conclusion is "A dynamic grid system (DGS) for providing privacy-preserving continuous Location Based Services (LBS). Dynamic Grid System (DGS) includes the query server (QS) and the service provider (SP), and cryptographic functions to split the complete query processing task into two parts that are performed separately by Query Server and Service Provider. Dynamic Grid System does not require any fully-trusted third party (FTTP); instead, we require only the much weaker assumption of no collusion among QS and SP. This separation also moves the data transfer load away from the user to the inexpensive and high-bandwidth link among QS and SP. We also designed efficient protocols for our DGS to support both continuous k-nearest-neighbor (KNN) and range queries. To evaluate the performance of DGS, we compare it to the state-of-the-art technique requiring a TTP. DGS provides better privacy guarantees than the Trusted Third Party scheme, and the experimental outcomes show that DGS is an order of magnitude more efficient than the TTP scheme, in terms of communication cost. In terms of computation cost, DGS also always outperforms the TTP scheme for Nearest Neighbor queries it is comparable or slightly more expensive than the TTP scheme for range queries.

REFERENCES

[1] B. Bamba, L. Liu, P. Pesti, and T.Wang, “Supporting anonymous location queries in mobile environments with PrivacyGrid,” in WWW, 2008.

[2] C.-Y. Chow and M. F. Mokbel, “Enabling private continuous queries for revealed user locations,” in SSTD, 2007.

[3] B. Gedik and L. Liu, “Protecting location privacy with personalized k-anonymity: Architecture and algorithms,” IEEE TMC, vol. 7, no. 1, pp. 1–18, 2008.

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

[5] 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.

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

[7] T. Xu and Y. Cai, “Location anonymity in continuous location-based services,” in ACM GIS, 2007.

[8] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, “Private queries in location based services: Anonymizers are not necessary,” in ACM SIGMOD, 2008.

[9] M. Kohlweiss, S. Faust, L. Fritsch, B. Gedrojc, and B. Preneel, “Efficient oblivious augmented maps: Location-based services with a payment broker,” in PET, 2007.

[10] C.-Y. Chow, M. F. Mokbel, and W. G. Aref, “Casper*: Query processing for location services without compromising privacy,” *ACM TODS*, vol. 34, no. 4, 2009.

[11] T. Xu and Y. Cai, “Feeling-based location privacy protection for location based services,” in *ACM CCS*, 2009.

[12] C.-Y. Chow, M. F. Mokbel, and X. Liu, “A peer-to-peer spatial cloaking algorithm for anonymous location-based service,” in *ACM GIS*, 2006.

[13] G. Ghinita, P. Kalnis, and S. Skiadopoulos, “MOBIHIDE: A mobile peerto- peer system for anonymous location-based queries,” in *SSTD*, 2007.

[14] “PRIVE: Anonymous location-based queries in distributed mobile systems,” in *WWW*, 2007.

[15] G. Zhong and U. Hengartner, “A distributed k-anonymity protocol for location privacy,” in *IEEE PerCom*, 2009.

[16] M. L. Yiu, C. S. Jensen, X. Huang, and H. Lu, “SpaceTwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services,” in *IEEE ICDE*, 2008.

[17] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, “Order-preserving encryption for numeric data,” in *ACM SIGMOD*, 2004.

[18] H. Hacig¨um¨us¸, B. Iyer, and S. Mehrotra, “Efficient execution of aggregation queries over encrypted relational databases,” in *DASFAA*, 2004.

[19] E. Mykletun and G. Tsudik, “Aggregation queries in the database-as-aservice model,” in *DBSec*, 2006.

[20] A. Khoshgozaran and C. Shahabi, “Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy,” in *SSTD*, 2007.

[21] W. K. Wong, D. W. Cheung, B. Kao, and N. Mamoulis, “Secure kNN computation on encrypted databases,” in *ACM SIGMOD*, 2009.

[22] M. L. Yiu, G. Ghinita, C. S. Jensen, and P. Kalnis, “Enabling search services on outsourced private spatial data,” *VLDB Journal*, vol. 19, no. 3, pp. 363–384, 2010.

[23] B. Palanisamy and L. Liu, “Mobimix: Protecting location privacy with mix zones over road networks,” in *IEEE ICDE*, 2011.

[24] S. Mascetti, C. Bettini, X. S. Wang, D. Freni, and S. Jajodia, “ProvidentHider: An algorithm to preserve historical k-anonymity in LBS,” in *MDM*, 2009.