iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 **Issue:** III **Month of publication:** March 2018

DOI: http://doi.org/10.22214/ijraset.2018.3613

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

User-content Fortification and Privacy Safeguarding for Location-based Queries

Ashwini S. Tagadghar[1], Srinidhi Chelmeda[2], M. Vidhya Sri[3], S.Avinash[4]
[1, 2, 3] *B. Tech, CSE, TKREC, Hyderabad*
[4] *M. Tech CSE, Assistant Professor TKREC, Hyderabad*

Abstract: ***The users are demanded incessantly to report their location to a potentially untrusted server to obtain services based on their location, which can subject them to privacy risks in Location Based Services (LBS). Alas, Prevailing privacy-conserving techniques have several disadvantages in LBS, such as demand for a FTT (i.e., fully trusted third) party, offering limited privacy guarantees and sustaining high communication overhead. In this paper, we propose a user-defined seclusion framework called Powerful Network Structure (PNS); the first universal system that fulfils four essential necessities for privacy-conserving snapshot and uninterrupted LBS. (1) This system only demands a STT (i.e. semi trusted third) party, responsible for carrying out simple matching operations correctly. This STT party does not have any information about a user's location. (2) Reliable snapshot and uninterrupted location privacy is guaranteed under our defined adversary models. (3) The communication cost for the user is only hinged on the number of relevant points of interests (POIs) in the vicinity of the user and not on his desired privacy level. (4) Although we only focus on range and k-nearest-neighbour queries in this work, our system can be easily extended to support other spatial queries without changing the algorithms run by the STT party and the database server, provided the required search area of a spatial query can be remote into spatial regions. Exploratory results report that our PNS is more efficient than the state-of-the-art seclusion-conserving technique for uninterrupted LBS.***
Keywords: location based services (LBS), power network structure (PNS), fully trusted third party (FTT), semi trusted third party (STT), privacy conserving, k-nearest neighbours, points of interests (POIs)

I. INTRODUCTION

The Location-Based Services(LBSs) have attracted a lot of attention in recent years. A typical example of LBSs is the search engine for the nearby Points of Interest (POIs). It helps the user look for POIs quickly and conveniently. To enjoy the LBSs, the query user should first submit his current location and query content (e.g. the description of POIs searched for) to the LBS Provider, then the LBS Provider searches for its database and returns the corresponding POIs to the query user. Although the LBSs bring great convenience to its users, for privacy and efficiency concerns, there are abundant problems that need to be solved before their real applications. First, the LBS Provider might be able to track the query user according to the user's location data. Second, the query content submitted by the query user could expose his sensitive information, such as personal interest. Third, the LBS Provider often has a lot of POIs that satisfy the query condition, but the query user would only be interested in a few POIs which are most relevant to his query. For example, when a user wants to find a ATM machine, he may intend to go to the nearest one; when a user wants to find a bar, he may wish to choose the most popular one. Therefore, devising a ranked query protocol, which preserves both the location privacy and query content privacy, is very important.

II. EXISTING SYSTEM

Spatial cloaking techniques have been widely used to preserve user location privacy in LBS. Most of the existing spatial cloaking techniques rely on a fully-trusted third party (TTP), usually termed *location* anonymizer that is required between the user and the service provider.

When a user subscribes to LBS, the location anonymizer will blur the user's exact location into a cloaked area such that the cloaked area includes at least k – 1 other users to satisfy k-anonymity.

In a system with such *regional location privacy* it is difficult for the user to specify personalized privacy requirements. The feeling-based approach alleviates this issue by finding a cloaked area based on the number of its visitors that is at least as popular as the user's specified public region. Although some spatial clocking techniques can be applied to peer-to-peer environments, these techniques still rely on the k-anonymity privacy requirement and can only achieve regional location privacy.

Furthermore, these techniques require users to trust each other, as they have to reveal their locations to other peers and rely on other peers' locations to blur their locations, another distributed method was proposed that does not require users to trust each other, but it still uses multiple TTPs.

Another family of algorithms uses incremental nearest neighbour queries, where a query starts at an "anchor" location which is different from the real location of a user and iteratively retrieves more points of interest until the query is satisfied. While it does not require a trusted third party, the approximate location of a user can still be learned; hence only regional location privacy is achieved.

III. PROPOSED SYSTEM

We propose a user-defined seclusion framework called Powerful Network Structure (PNS) 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 PNS, is that, 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 powerful network structure specified by the user. Then, the user encrypts a query that includes the information of the query area and the powerful network 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.

The main contributions of this work are:

A. We propose to preserve both the location privacy and the query content privacy in the LBSs.
B. We systematically construct a secure query protocol, where different data providers can use different secret keys to encrypt their data, the query users can generate query conditions without knowing these keys and query over these data securely and efficiently.
C. We devise a ranked and fine-grained query protocol, where query users can query multiple POIs according to their interests and obtain the most relevant data according to their evaluation criteria.
D. We give rigorous security analysis and conduct extensive experiments on the real-world data sets, which confirms the efficacy and efficiency of our proposed schemes.

IV. ARCHITECTURE

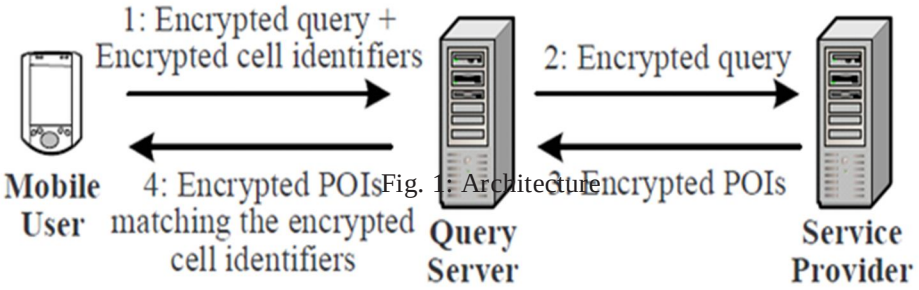


Fig. 1: Architecture

In, this architecture, when a mobile user sends a query, the query and the cell identifiers are encrypted by the mobile and sent to the query server. The query server stores the cell identifiers in encrypted form and forwards encrypted query to the service provider. The main purpose of the query server is to match the user's cell identifiers and the cell identifiers sent by the service provider. The service provider decrypts the encrypted query and finds the relevant result for the query provided by the user from his database and retrieves resultant POIs. These resultant POIs are forwarded to the query server in an encrypted form along with

some cell identifiers which are also encrypted. Then the query server performs its matching operation of encrypted cell identifiers of both the user and the service provider and upon match being found, it forwards the encrypted POIs to the mobile user. The mobile decrypts the encrypted POIs and provides the required location result.

V. ALGORITHM

The algorithm for communication between the mobile user, the query server and the service provided is as follows:
The user uploads his query in an encrypted format in the query grid (QG1). Here y_1 is the public key for the row and x_1 is chosen at random and y_2 is public key for the column and x_2 is chosen at random. The variable C_1 stores the encrypted query as A_1 and encrypted cell identifiers as B_1 for the the User1 and similarly for User2....User n, the variables C_2,….C_n are used. These variables C_1, C_2…… C_n are sent to the query server. The query server takes the Request Grid (RG1) as the input containing the variables C_1…..C_n and stores the encrypted cell identifiers and transfers the encrypted query to the service provider as $C'_{1,\alpha}$….$C'_{n,m}$ and performs matching of encrypted cell identifiers of the mobile user and the service provider and upon matching forwards the required result to the user.

Input: User: i, j
Output: User: $(ID_{Q_{i,j}}, k_{i,j})$
1: **User** $(QG1)$
2: $y_1 \leftarrow g_1^{x_1}$, where y_1 is the public key for the row and x_1 is chosen at random
3: $y_2 \leftarrow g_2^{x_2}$, where y_2 is the public key for the column and x_2 is chosen at random
4: $\boldsymbol{C_1} \leftarrow (A_1, B_1) = (g_1^{r_1}, g_1^{-i} y_1^{r_1})$
5: $\boldsymbol{C_2} \leftarrow (A_2, B_2) = (g_2^{r_2}, g_2^{-j} y_2^{r_2})$
6: $Server \Leftarrow \boldsymbol{C_1}, \boldsymbol{C_2}$
7: **Server** $(RG1)$
8: $\boldsymbol{C'_{1,\alpha}} \leftarrow (A_1^{r'_\alpha}, g_1^{R_\alpha} r_R (g_1^\alpha B_1)^{r'_\alpha})$ for $1 \leq \alpha \leq n$ and $r_R = g_1^s$, where s is chosen randomly
9: $\boldsymbol{C'_{2,\beta}} \leftarrow (A_2^{r'_\beta}, g_2^{C_\beta} r_C (g_2^\beta B_2)^{r'_\beta})$ for $1 \leq \beta \leq m$ and $r_C = g_2^t$, where t is chosen randomly
10: $\gamma \leftarrow g_0^{1/r_R r_C}$
11: $User \Leftarrow \boldsymbol{C'_{1,1}}, ..., \boldsymbol{C'_{1,n}}, \boldsymbol{C'_{2,1}}, ..., \boldsymbol{C'_{2,m}}, \gamma$

Fig. 2: Transfer Algorithm

VI. MODULES

The modules that are considered to have much significance in this are as follows:

A. Service Providers

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 our system architecture, SP does not communicate with mobile users directly, but it provides services for them indirectly through the query server (QS).

B. Mobile Users

Each mobile user is equipped with a GPS-enabled device that determines the user's location in the form. 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. Then, 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.

C. Query Server

QS is a semi-trusted party placed between the mobile user and SP. Similar to the most popular infrastructure in existing privacy-preserving techniques for LBS, QS can be maintained by a telecom operator.

VII. SECURITY ANALYSIS

We analyze the security of our scheme. In what follows, we first demonstrate that query content privacy is well preserved. Then we prove that the location privacy is preserved we analyze the query content privacy of our scheme from three aspects:

A. Query content privacy analysis

First, let's consider the following game played between a LBS Provider, denoted by A, and a Challenger, which acts as the query user and Data Provider. generates parameters k, r, g, secret hash function $H_s(\ )$ and a keyword d_i^j, sends q, $q^{k \cdot r \cdot H_S(d_i^j)}$, $q^{k \cdot r}$ to A. Based on these information, A would try to calculate d_i^j. However, the Discrete Logarithm Problem is hard, it is intractable to compute d_i^j in polynomial time. Besides, each POI record stored on the LBS Provider is accompanied by an expiration time, namely, these data will be useless once the actual time exceeds the expiration time. Therefore, the security of keywords submitted by the Data Providers and query users are well preserved.

Second, since the LBS Provider is also responsible for ranking and returning query results, the LBS Provider would try to deduce sensitive information based on these query results. Specifically, the LBS Provider would go to a specific place, obtain the corresponding POI there, know about what is at the location and further guess the query content of the query user. However, this attack is infeasible. The reason is demonstrated as follows, in our scheme, to prevent the LBS Provider from knowing the query content submitted by the query user, we propose to insert disturbed keywords before submitting query content to the LBS Provider, obviously, the LBS Provider cannot distinguish which keywords are query content of the query user.

Third, the LBS Provider would want to disguise as a query user to submit query keywords for all types of POI, and exhaustively tests whether a query keyword matches with a POI stored on it. However, this attack is also inapplicable. The reason is that, in our scheme, only the Data Provider and the valid query user knows the secret hash function $H_s(\ )$. Without knowing this secret hash function, the LBS Provider cannot even generate the correct query conditions. Therefore, our scheme can efficiently defend against this attack.

B. Location privacy analysis

We analysis the location privacy of our scheme. Recall that, instead of sending his exact location or region of interest to the LBS Provider, the query user generates a concealing disk fully covering his region of interest and sends it to LBS Provider. We denote this concealing disk by $CD = ((x_c, y_c), r_c)$, where (x_c, y_c) is the center of the concealing disk, r_c is the radius of the concealing disk. The LBS Provider only learns that the query user is located in the concealing disk. When we combine our scheme with some anonymity techniques, the privacy can be further enhanced.

To further analyze the location privacy of our scheme, we consider the following game played between an adversary A and a Challenger C which acts as the query user:

C generates parameters r_c, (x_c, y_c), θ, and d_{RC}, sends r_c, (x_c, y_c) to A. A restores the query area based on r_c, (x_c, y_c) he received, A wants to narrow down the area the user located, and learns that the query user is located on the circumference of a circle, whose radius is d_{RC} (the distance between query user's real location and the center of concealing disk (x_c, y_c)), as shown by the dotted line circle in Fig. 3. However, since d_{RC} is randomly generated by C and is only known to C, A might only be able to get d_{RC} by making a random guess between 0 to r_c. Thus, the probability that the LBS Provider finds out query user's exact location is $1/\sqrt{\pi r_c}$

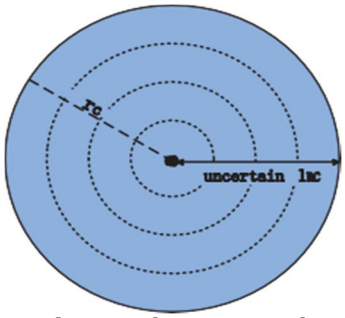


Fig. 3: deducing user's exact location by guessing the value of l_{RC} from 0 to r_c

VIII. RESULT ANALYSIS

A. Simulation Settings

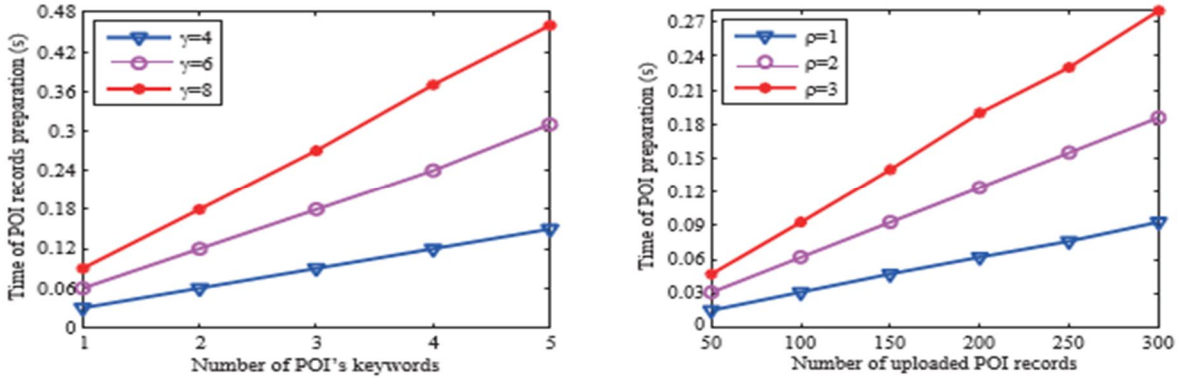
The experiment programs are coded using Python programming language on a PC with 3.0GHZ Pentium Dual Core CPU and 2GB memory. In all experiments, we randomly generate 50 users, 3 Data Providers and 1000 POIs into 8km area. For every 5 minutes, 10% of users are randomly selected to issue queries. We implement all necessary routines for Data Providers to pre-process POI records, for query user to generate query conditions, and for the LBS Provider to perform the queries.

B. Time cost of preparing POIs

Fig. 4 demonstrates the time cost of preparing POI records, we test the efficiency with different average number of keywords ρ of each POI records, and different number of POI records γ required to be uploaded to the LBS Provider, respectively. From Fig. 4(a), we can observe that the time of preparation POI records increases as ρ increases. From Fig. 4(b), we observe that, with the γ increases, the time cost of preparing POI records also increases linearly, which is acceptable because of the number of records added finitely in each Data Providers.

C. Time cost of encrypting query keywords

Fig. 5 illustrates the time cost of generating query conditions. We evaluate the relationship between the time cost of encrypting keywords and the varying number of a user's key- words μ. This experiment is meaningful because the number of a user's keywords is closely related to the time cost. The simulation results are shown in Fig. 5, we can see that, when the μ increases from 3 to 9, the time cost of encrypting keywords increases from 0.93ms to 2.97ms, which is viable due to query user only submit very seldom query keywords.



(a) For different average number of keywords of each POI.

(b) For different number of POI records.

Fig. 4: Time consumption of POI preparation.

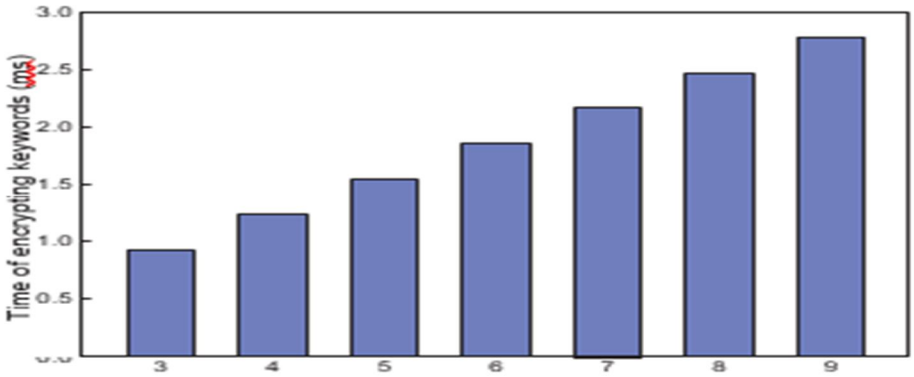


Fig. 5: Time cost of encrypting query keywords

IX. CONCLUSION

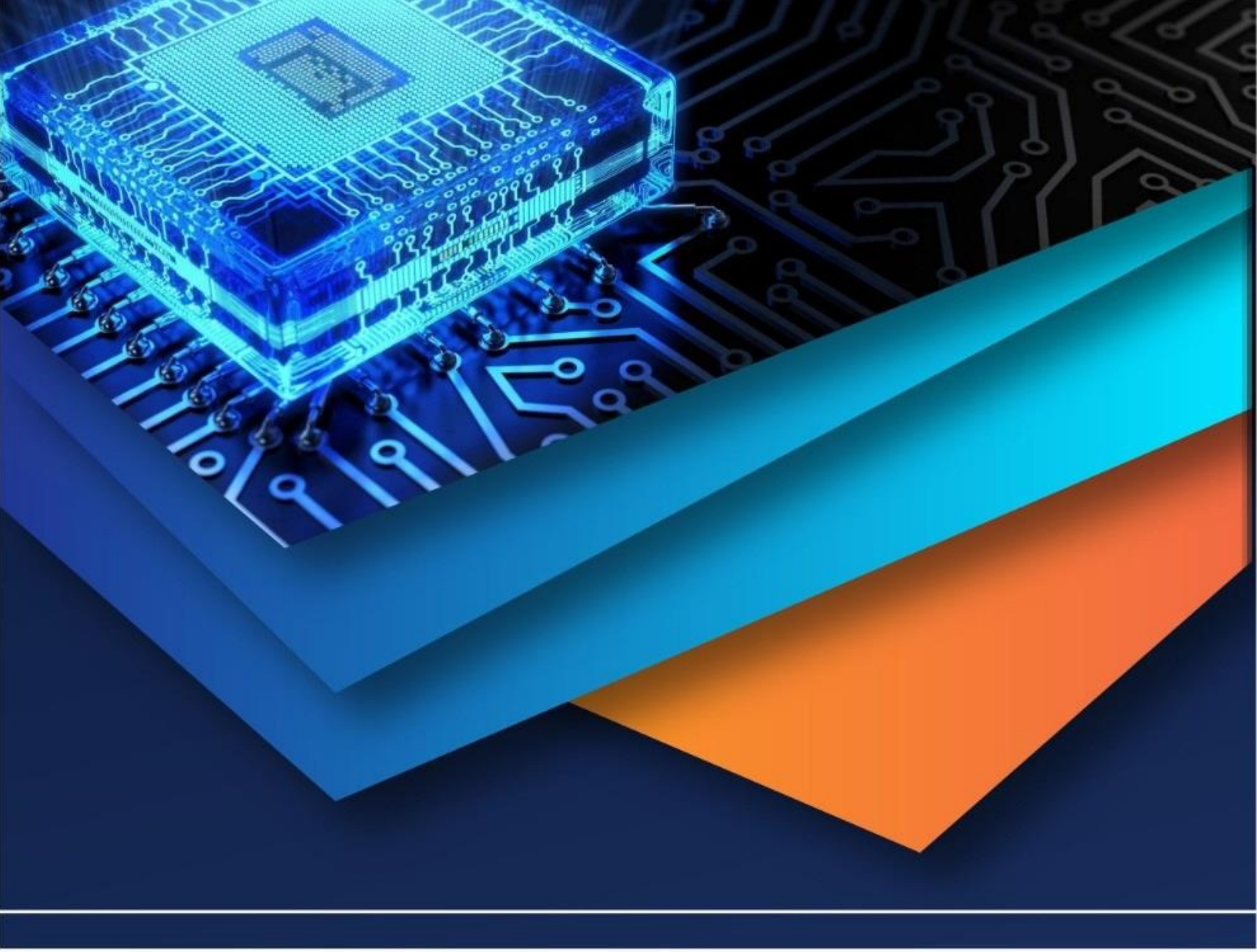
We propose a ranked query protocol in the Location-based Services, while preserving both the location privacy and content privacy. First, we propose to hide the query users' region of interest from the LBS Provider, which prevents the LBS Provider from knowing the exact location of the LBS users. Second, we systematically construct a secure query protocol, where different Data Providers can use different secret keys to encrypt their POI data, the users can query over these data securely and efficiently. These designs prevent the LBS Provider from deducing the content of query data. Third, we devise a ranked search protocol, which is compatible with many widely used search algorithms. Finally, we give rigorous security analysis and conduct extensive experiments, which confirms the efficacy and efficiency of our proposed scheme.

X. FUTURE ENHANCEMENTS

Future work can be done by enhancing to file privacy using semi trusted third party (STTP). These challenges need to be furthermore focused to achieve full potential privacy conserving management.

REFERENCES

[1] Niu, Q. Li, X. Zhu, G. Cao, and H. Li, Enhancing privacy through caching in location-based services in Proc. of IEEE INFOCOM, 2015.
[2] R. Lu, X. Lin, Z. Shi, and J. Shao, Plam: A privacy-preserving framework for local-area mobile social networks in INFOCOM, 2014 Proceedings IEEE, 2014, pp. 763–771. M. Li, S. Salinas, A. Thapa, and P. Li, n-cd: A geometric approach to preserving location privacy in location-based services in INFOCOM, 2013 Proceedings IEEE. IEEE, 2013, pp. 3012–3020.
[3] Y. Wang, D. Xu, X. He, C. Zhang, F. Li, and D. Xu, L2p2: Location- aware location privacy protection for location-based services in INFO- COM, 2012 Proceedings IEEE. IEEE, 2012, pp. 1996–2004.
[4] Pingley, N. Zhang, X. Fu, H.-A. Choi, S. Subramaniam, and W. Zhao, Protection of query privacy for continuous location-based services in INFOCOM, 2011 Proceedings IEEE. IEEE, 2011, pp. 1710–1718.
[5] Q. Zhang and L. Lazos, Collision-resistant query anonymization for location-based services in Communications (ICC), 2014 IEEE International Conference on. IEEE, 2014, pp. 768–774.
[6] Niu, X. Zhu, W. Li, and H. Li, Epcloak: An efficient and privacy- preserving spatial cloaking scheme for lbss in Mobile Ad Hoc and Sensor Systems (MASS), 2014 IEEE 11th International Conference on, 2014, pp. 398–406.
[7] J. Shao, R. Lu, and X. Lin, Fine: A fine-grained privacy-preserving location-based service framework for mobile devices Proceedings - IEEE INFOCOM, pp. 244–252, 2014.
[8] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, Privacy-preserving multi-keyword ranked search over encrypted cloud data Parallel and Distributed Systems, IEEE Transactions on, vol. 25, no. 1, pp. 222–233, 2014.
[9] W. Zhang, S. Xiao, Y. Lin, J. Wu, and S. Zhou, Privacy preserving ranked multi-keyword search for multiple data owners in cloud computing IEEE Transactions on Computers, 2015.
[10] R. Schlegel, C.-Y. Chow, Q. Huang, and D. Wong, User-defined privacy grid system for continuous location-based services.
[11] W. Zhang, Y. Lin, S. Xiao, Q. Liu, and T. Zhou, Secure distributed keyword search in multiple clouds in Proc. IEEE/ACM IWQOS'14, Hongkong, May 2014, pp. 370–379.
[12] J. W. Byun, H. S. Rhee, H.-A. Park, and D. H. Lee, Off-line keyword guessing attacks on recent keyword search schemes over encrypted data in Secure Data Management. Springer, 2006, pp. 75–83.
[13] Boneh and M. Franklin, Identity-based encryption from the well pairing SIAM Journal on Computing, vol. 32, no. 3, pp. 586–615, 2003.
[14] R. Beresford and F. Stajano, Location privacy in pervasive computing IEEE Pervasive computing, no. 1, pp. 46–55, 2003.
[15] Wang, N. Cao, J. Li, K. Ren, and W. Lou, Secure ranked keyword search over encrypted cloud data in Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on. IEEE, 2010, pp. 253–262.
[16] M. Gruteser and D. Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking in Proceedings of the 1st international conference on Mobile systems, applications and services. ACM, 2003, pp. 31–42.
[17] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, Preventing location-based identity inference in anonymous spatial queries IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 12, pp. 1719–1733, 2008.
[18] Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramani- am, l-diversity: Privacy beyond k-anonymity ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, no. 1, p. 3, 2007.
[19] R. Beresford and F. Stajano, Mix zones: User privacy in location- aware services 2004.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY