ARTICLE IN PRESS

Future Generation Computer Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

ASA: Against statistical attacks for privacy-aware users in Location Based Service

Yanming Sun [a], Min Chen [a], Long Hu [a,*], Yongfeng Qian [a], Mohammad Mehedi Hassan [b,c]

[a] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China

[b] Computer Engineering Department, King Saud University, P.O. Box 51178, Riyadh, Saudi Arabia

[c] Information Systems Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

HIGHLIGHTS

- According to the rule of activity of most users, we introduce LSA by using the historical data. For the attack, we give out two methods to preserve user's privacy.
- We divide the regions in the map into different PLs according to the privacy requirement. We design algorithm to make the regions of high level to be dummies at a high rate and the regions of low level at a low rate. The problem that the attacker can violate the privacy of a particular region by analyzing the historical data is solved.
- We analyze the ability to preserve user's privacy by entropy. The internal relation among the frequency of user's LBS query, the division of regions in the map, and the length of the interval of historical information collected is discussed.

ARTICLE INFO

Article history:
Received 31 August 2015
Received in revised form
14 March 2016
Accepted 19 June 2016
Available online xxxx

Keywords:
Location privacy
Privacy preserving
k-anonymity
Information entropy
Location based service (LBS)

ABSTRACT

The fusion of mobile devices and social networks is stimulating a wider use of Location Based Service (LBS) and makes it become an important part in our daily life. However, the problem of privacy leakage has become a main factor that hinders the further development of LBS. When a LBS user sends queries to the LBS server, the user's personal privacy in terms of identity and location may be leaked to the attacker. To protect user's privacy, Niu et al. proposed an algorithm named enhanced-Dummy Location Selection (en-DLS). In this paper, we introduce two attacks to en-DLS, namely long-term statistical attack (LSA) and regional statistical attack (RSA). In the proposed attacks, an attacker can obtain the privacy contents of a user by analyzing LBS historical data, which causes en-DLS to be invalid for user's privacy protection. Furthermore, this paper proposes a set of privacy protection schemes against both LSA and RSA. For LSA, we propose two protection methods named multiple user name (MNAME) and same user name (SNAME). To solve the regional privacy issue, we divide the map into various regions with different requirements on privacy protection. For this purpose, four levels of protection requirements (PLs) are defined, and true location is protected by allocating a certain number of positions from the dummies according to the location's PL. Performance analysis and simulation results show that our proposed methods can completely avoid the vulnerabilities of en-DLS to both LSA and RSA, and incur marginal increase of communication overhead and computational cost.

1. Introduction

With the development of mobile computing and network technology, mobile phone has become a necessity in people's life. Besides satisfying the need of daily communications, mobile phone also provides many convenient services for human being [1]. With the development of smartphones, Global Positioning System (GPS) has been solidified in the most smartphones and provides function for mobile service provider to position the smartphone. America E-911 document has pointed out that mobile service providers should provide location recognition service in 125 m in order that the owner of mobile phone can obtain timely rescue and help when she is in danger, such as fire or hijacked [2].

* Corresponding author.
E-mail addresses: yanming.epic@gmail.com (Y. Sun), minchen2012@hust.edu.cn (M. Chen), longhu.cs@gmail.com (L. Hu), yongfengqian.epic@qq.com (Y. Qian), mmhassan@ksu.edu.sa (M.M. Hassan).

http://dx.doi.org/10.1016/j.future.2016.06.017
0167-739X/

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

Recently, many Location Based Service (LBS) applications come into being. In spite of various benefits brought by LBS, the intrinsic privacy leakage problem cannot be ignored due to the openness of wireless networks [3]. At present, privacy leakage issues become the main obstacle to the wide application of LBS services.

As stated in [4], location privacy is "*the ability to prevent others from obtaining the current or past location of the user*". For privacy preserving in LBS service, there are several challenges:

- High Precision: The user's identity and location should be protected. Meanwhile, the precision of LBS service should be ensured.
- Low Overhead: The communication, computation and storage ability of the user terminal is limited. Thus, the communication overhead, the computational cost, and the storage overhead should be low in preserving the user's privacy.
- Privacy: LBS server itself may be an attacker. It can obtain the user's real location and historical data directly.

The main solutions to protect user's privacy can be divided into obfuscation and anonymity according to the technology used. In anonymity, using dummy is efficient since it need not a trusted third party to preserve privacy and it attracts many scholars' attention. Among them, Niu et al. proposed enhanced-Dummy Location Selection (en-DLS) based on the probability of sending LBS queries from a location in the history by users [5]. It solves the problem of privacy leakage in a single LBS query. In [6], Niu et al. proposed Caching-aware Dummy Selection Algorithm (CaDSA) and enhanced-CaDSA which use caching to improve the privacy of user. En-DLS has the following characteristics:

- Side information: In en-DLS, side information refers to the terrain information in the city. The dummies are not selected from the rivers or mountains in the city, but carefully selected based on the historical query probability in the locations. The problem of reducing in protection caused by side information is solved.
- Cloaking area: To overcome the disadvantage of k-anonymity, in en-DLS, the coverage area of the dummies is selected as large as possible.
- Implementation issues: In en-DLS, accessing to the historical queries is fully considered. Access Points (AP) based method is proposed. The communication overhead is relational.

Although en-DLS solved the problem of privacy leakage in a LBS query, it has vulnerabilities. In this paper, we introduce two attacks to en-DLS, namely long-term statistical attack (LSA) and regional statistical attack (RSA). The attacker can obtain user's privacy contents using historical statistics. For an attacker, after compromising the LBS server, he can obtain a large number of historical data. We introduce an attack named LSA to obtain user's real identity and location using these historical data. We study based on LSA and put forward two methods to preserve privacy named multiple user name (MNAME) and same user name (SNAME). Besides LSA, the attacker can obtain the historical LBS applications from a particular region. Furthermore, the attacker can obtain a lot of information about the user from the region through statistics. For this problem, we propose a method to divide the regions in the map into different privacy levels (PLs). Then, we delete some dummy locations in en-DLS and select some locations from high PL regions to protect the privacy of the regions. We take entxu2007preventingabbas2013collusionropy as the metric to analyze the ability of the proposed methods. The performance analysis and simulation results show that the proposed methods can effectively preserve user's privacy against LSA and RSA. The main contribution of this paper includes the following aspects:

- According to the activities of most users, we introduce LSA. For the attack, we give out two methods to preserve user's privacy.
- We divide the regions in the map into different PLs according to the privacy requirement. We give out an algorithm to make the regions of high PL to be dummies at a high rate and the regions of low PL at a low rate. The problem that the attacker can violate the privacy of a particular region by analyzing the historical data is solved.
- We analyze the ability of preserving user's privacy by entropy. The relation among the frequency of user's LBS query, the division of regions in the map, and the length of the interval of historical information collected is discussed.

The rest of this paper is organized as follows. Section 2 gives out some preliminaries and motivation of this paper. In this section, we give out LSA and RSA. In Section 3, we propose methods to resist LSA and RSA. In Section 4, we discuss the security and performance of the proposed methods. Section 5 presents the simulations. In Section 6, we review the related work. Conclusion and future work are in Section 7.

2. Preliminaries

In this section, we first introduce the privacy metrics and attack model. Then, we give out the motivation of our solution.

2.1. Metrics for privacy

To measure the ability of preserving privacy, we need some metrics. There are five kinds of metrics currently [7]. They are uncertainty-based metric, "clustering error"-based metric, traceability-based metric, k-anonymity metric, and distortion-based metric. In this paper, we use uncertainty-based metric to measure privacy in communication system. In [8], the author put forward to measure the ability of an attacker by differentiating the real locations from the anonymous set. The author pointed out *k-anonymity* is really achieved if the attacker cannot distinguish the real location from the $k-1$ locations in the same transmission. In [5], the author proposed that the direct method to measure the privacy preserving ability in *k-anonymity* is to use the k. The larger k denotes the higher ability to preserve privacy. However, there are some disadvantages in this measurement. For example, the $k-1$ dummies may be selected in the rivers, lakes, mountains, or in the impossible positions to reach in the path for the limited of speed. The attacker can easily distinguish them as unlikely LBS query positions from the real location. Therefore, simply using k as the metric cannot express the ability of privacy protection accurately. Besides k, entropy is widely used to measure the ability [5,6,8–10]. Entropy is first used to measure privacy in [11]. As we all know, entropy is often used to measure the uncertainty of a system. In privacy protection, entropy can be used to measure the degree of uncertainty of a location belonging to a user. In *k-anonymity*, from the point of view of the attacker, in the anonymous set consists of the real location and $k-1$ dummies, the probability of a location to be the real one is p_i. In the anonymous set, the sum of all probabilities p_i is one. Thus, the entropy H of identifying a real location in the candidate set is

$$H = -\sum_{i=1}^{k} p_i \cdot \log_2 p_i. \tag{1}$$

When all the k locations in the set have the same probability, the maximum entropy is achieved, where the probability p_i is $1/k$ for all the locations and the maximum of H is $\log_2 k$.

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

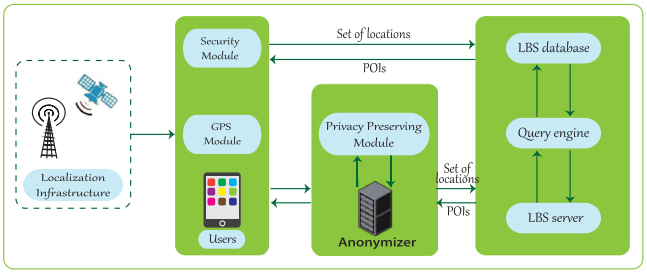

Fig. 1. Structure of LBS system.

2.2. Adversary model

In this paper, we assume that the users' accesses to the LBS are sporadic, which means the period between two successive LBS application cannot be neglected. As in [5], in this paper, we assume that the format of user's applying information is $\langle (x, y), I, r,$ others$\rangle$. Among them, (x, y) refers to the location of the user. I indicates the interest of the user, that is the type of LBS requests. The range is r. Others include user's identity and other information.

The goal of the attacker is to obtain the user's privacy information, including name, interest, and location. An attacker could monitor around to obtain user's applying packets and obtain private information sent by the user. An adversary may also monitor a user to crack the points of interest (POIs) sent by LBS server to the user. Then he can infer the user's identity, location, interest, etc. An attacker can also directly compromise the LBS server to obtain the historical data of users. In this paper, we assume LBS is the attacker. For commercial purposes, he attempts to obtain information related to user's privacy. He interests in user's real location and type of LBS query. He cannot only obtain current user's LBS query, but also the historical data of the user. He also knows the privacy protection mechanism.

2.3. Motivation and new approach

When a user sends a LBS query by his smartphone, the location of the smartphone is first determined by GPS service. Then, the smartphone forwards the user's identity, location, interest, and the range of the query to the LBS server either directly or in-directly through anonymizer. At last, LBS server will reply according to the user's query and feedback POIs as shown in Fig. 1.

As stated above, the attacker can obtain the same information of users as LBS server. In traditional privacy preserving method, $k - 1$ dummies are selected randomly to confuse the attacker and protect the real location. However, in [5], the author found that because of the different layout of terrain and living area, the probabilities of applying LBS service in the regions are different. For example, in some city, there are rivers or mountains. Users can hardly be in these regions to apply LBS service. So the traditional method to select $k - 1$ dummy locations cannot protect the real location efficiently because of the side information. For example, when k is 20, if 14 dummies in 19 dummy locations are selected in the regions with low probabilities of applying LBS service, the attacker can easily filter out the 14 dummies. In [5], the author quantified this problem using entropy and proposed DLS which selects $k - 1$ dummies from the grids with the same or similar probabilities. The DLS algorithm is shown in Fig. 2.

The author simplified the regions in the city to a grid. Based on the historical data, the author used different shades in the grids to indicate different probabilities of user's applying LBS services in the regions. Blank squares represent users never or rarely send LBS queries from the regions in the past. In the figure, candidate cells

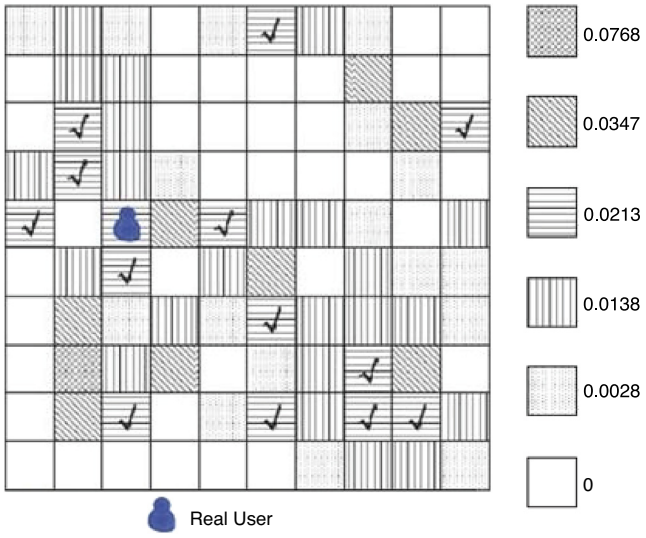

Fig. 2. DLS algorithm.

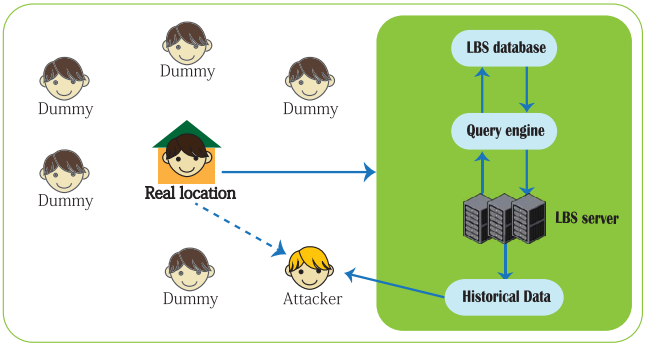

Fig. 3. Attack through using the historical data.

to be selected as dummies which has similar probability as the real location are marked with $\surd$.

Based on DLS, the author proposed en-DLS which makes the cloaking area of dummies as large as possible.

Although en-DLS solves the problem of privacy leakage in a LBS query, it has the following vulnerabilities. Refer to Fig. 3, assume Bob often sends LBS queries in his home. The real location will be disclosed to the attacker due to the accumulation of historical data. It is because in en-DLS, for every LBS application, dummies are decentralized, whereas the real locations are concentrated relatively. The attacker can obtain the user's real location from the historical data. Moreover, he can obtain the privacy contents about the user.

In en-DLS, assume k is 20. Then, $k - 1$ is 19. When a user applies LBS, 19 dummy locations will be carefully selected to protect the real location. This method can efficiently protect user's privacy in one service. However, considering the behavioral pattern in people's daily life, most activity places where a user stays mainly concentrate in the home, the place of work, and the fixed entertainment place (such as fixed cinema or cafe). So, an attack is proposed as follows. An attacker first captures and analyzes a user's historical data. Through looking for the LBS request of the particular user in a certain period of time, even if every request is protected by 19 carefully selected dummy locations, the attacker is still able to obtain the main locations of LBS requests by analyzing the historical data after the amount of privacy data are accumulated to a certain extent. Then, the attacker can deduce the user's identity, the place of work, personal interest, and other privacy contents from the obtained data.

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

Assume the attacker has obtained the historical LBS application data of user S. The time interval of the first LBS request to the last LBS request of S in the data is t. S sends LBS queries in the frequency of q. The map is divided into $r \times r$ grids. The en-DLS in [5] is adopted. Each time the user applies LBS service, there will be $k-1$ selected dummies. For simplicity, we assume that the user only applies LBS service at home. Assume there are m grids that have the same probabilities as the location of S and can be selected as dummies. Then there is the following relationship. Every time S sends a LBS query, any of the m grids has probability p_m to be a dummy location.

$$p_m = \frac{C_{m-1}^{k-2}}{C_m^{k-1}} = \frac{k-1}{m}. \tag{2}$$

In the time interval t, user S has applied LBS services for n times.

$$n = t \cdot q. \tag{3}$$

In these applications, $k \times n$ locations are generated. Among the locations, the number of real location is n and the location is user's home. The rest of $(k-1) \times n$ locations are dummies. In m, every grid has n_d times to be dummies in $(k-1) \times n$.

$$n_d = n \times p_m = \frac{n \times (k-1)}{m} \tag{4}$$

$$n - n_d = n \times \left(1 - \frac{k-1}{m}\right). \tag{5}$$

We assume that every time user S applies LBS service, there are enough grids to be dummies. That is to say, m is far large than $k-1$. Assume $k-1 > 0$. We can conclude that $p_m < 1$ and $n_d < n$. After calculation, we get:

$$\lim_{m\to\infty} p_m = \lim_{m\to\infty} \frac{k-1}{m} = 0. \tag{6}$$

We draw the following conclusion: the bigger the m, the more is n-n_d. That is to say, as long as m is large enough, the times of every grid in m to be dummy location will be close to 0. Because n is the times of user S applying LBS service and it is a constant. So, as long as n is large enough (that is to say the user's historical data of LBS application is enough), we can conclude that the times of LBS application in real location will be far more than in any dummy location. This shows that if the real locations of the user are relatively concentrated, the locations of dummies are relatively decentralized because of their randomness and the dummies can be ignored by the attacker. So the real location cannot be protected by these dummies efficiently. It needs to be explained that in this paper, we assume the frequency of user applying LBS service q as a determined value. In reality, user applies LBS service in a random manner. For simplicity, in the latter part of this paper, q will also be regarded as determined. Analysis on the reality of a random q can be the work of future. In the above method, attackers can obtain privacy contents by analysis of historical data. In this paper, we call this kind of attack long-term statistical attack (LSA).

Below, we will illustrate the attack by experiment.

Fig. 4 is a statistical map of user Bob applying LBS services in a month. Bob applies LBS service about 3 times a day. The map is divided into 10×10 size of grids. Each point in the grid represents the LBS server received one LBS application from the location. In these applications, some applications are from real locations, others are from selected dummies. Among them, the number of Bob's real LBS applications is 90. Every time Bob sends a real application, 19 dummy locations will be generated at random. To illustrate the problem, we select dummies at random in the experiment. The approach to select dummies according to en-DLS in [5] is similar as this.

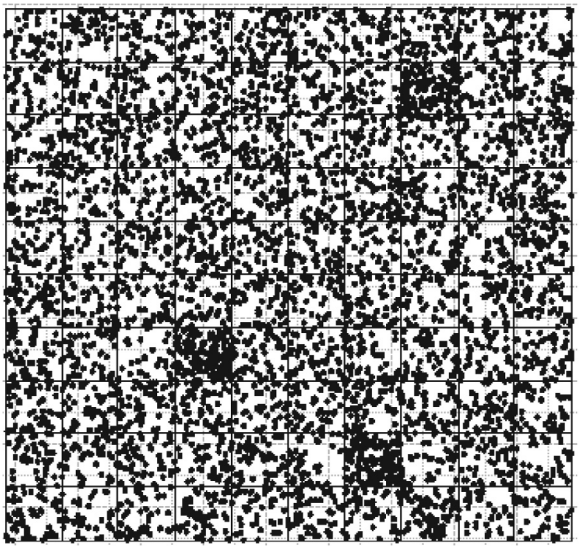

Fig. 4. Long-term statistical attack.

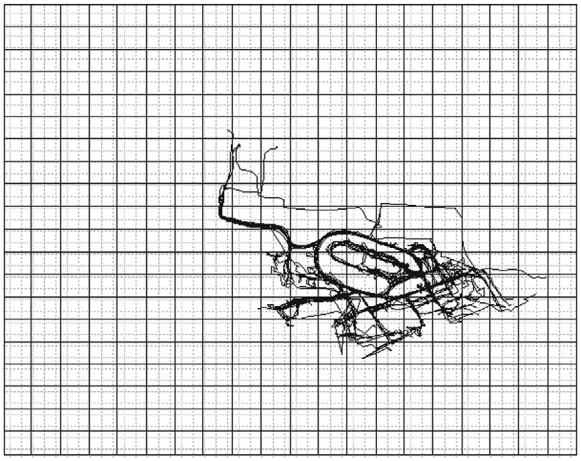

Fig. 5. User's real trajectory.

As we can see from the figure, the majority of user's applying locations are in three grids. The three grids are respectively corresponding to Bob's home, company and cafe where Bob spends his leisure time. According to Fig. 4, the attacker can obtain Bob's real locations by statistics and then infers Bob's privacy, such as identity, place of work and personal habit.

Fig. 5 is the real applying locations of a user in a day. The user applies LBS service every 30 s. Fig. 6 is the results of generating 19 dummies for every real application. From the figure, we can see although dummies are added, the locations where the user's real LBS applications concentrate cannot be protected efficiently.

Besides LSA, there is another vulnerability in en-DLS in privacy protection. The attacker can collect and analyze LBS queries in a particular region according to the historical data. If the region is corresponding to Bob's home and in a long period, the applications from this region are mainly about health [12], the attacker can deduce that the user in that region has problem with his health even if the method in [5] is used. If Bob lives alone, the attacker can obtain that the user who lives in the region is Bob by social engineering and he can know Bob has health problem. The attacker can sell this information to commercial institutions and Bob will be rather baffling to be sent a large number of heath care advertisement. Bob will feel his privacy has been violated. If Bob lives with his family, even if it is unable to accurately determine who has health problem and unable to determine the identity

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

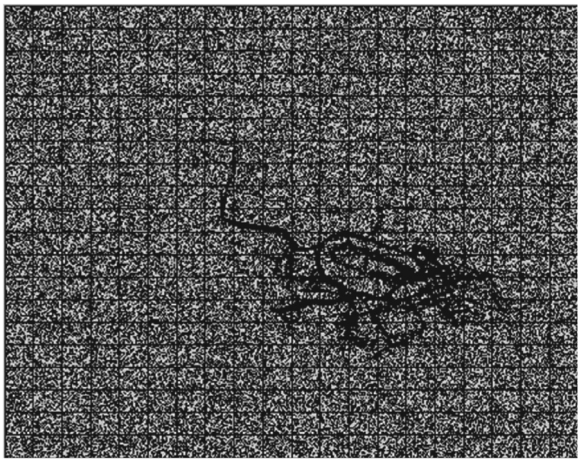

Fig. 6. Results with dummies selected at random.

of the user, the attacker can also draw the conclusion that there is someone has health problem in that family. Thus, the family's privacy has been violated.

Another example is in a secret department (such as government agency, military industrial enterprise) there are a large number of employees. The attacker analyzes the LBS queries in the historical data from the region. Assume in a certain period of time, at 5:00 in the afternoon, most of the staff in the department go off work and they apply traffic information through LBS service. Even dummies are taken, through historical data analysis, the attacker can know that at 5:00 in the afternoon there are many users who apply traffic service. Then, the attacker can conclude that the department stops working at 5:00 in the afternoon. If in a period of time, the attacker obtains from the historical data that most users send traffic queries at 6:00 in the morning, he can easily conclude that in the period of time, the department works overtime at night. He can also speculate that in this period of time, the department is engaged in an important secret project. According to the similar traces, the attacker can obtain more or less information associated with the work of the department. Then, the secrets of the department will be violated.

On the other hand, if the map is divided into many grids, using dummy to achieve privacy preservation may lead to the sensitive region has less probability to be dummies of other LBS queries. Therefore, less application will be sent from the region. Through historical data analysis, the attacker can easily obtain real application from the region.

Based on the method in [5], LBS server is the attacker. The map is divided into $r \times r$ grids. Assume in grid A, only user S applies LBS service. In the time interval of t, from the view of the attacker, user S in grid A applied LBS service for s times. $s = s_{real} + s_{dummy}$, where s_{real} denotes the times of real LBS applications from the user in A in the time interval of t and s_{dummy} denotes the times which A is taken as dummy of other LBS applications in t. Assume there are m grids which has same or similar probability as A. That is to say, when S in A applies LBS service, m locations are available for selection to be dummies. In turn, when a user in any of the m locations applies LBS service, A may be a dummy. Assume there are n users in all of the m locations and the frequency of the LBS application from every user is q. When a user applies LBS service, $k-1$ dummies are generated. Then, in the period of t, the number of total LBS applications in the m locations is $n \times t \times q$. In one application, the probability of A to be a dummy, p_A, is:

$$p_A = \frac{C_{m-1}^{k-2}}{C_m^{k-1}} = \frac{k-1}{m}. \tag{7}$$

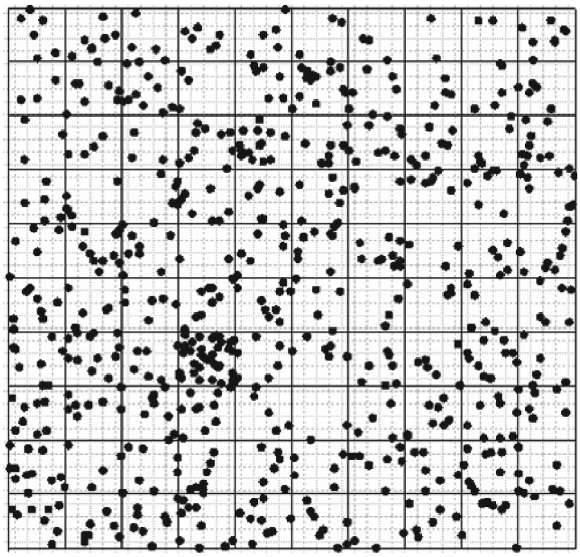

Fig. 7. Regional statistical attack.

In the period of t, the total times of A to be dummy of others, s_{dummy}, is:

$$s_{dummy} = p_A \cdot n \cdot t \cdot q = \frac{ntq(k-1)}{m}. \tag{8}$$

We can see from (8) that bigger is m, smaller is s_{dummy}. If s_{real} is big enough, the LBS applications from A can be regarded as real application. Because LBS server is the attacker, he can easily obtain k and compute s_{dummy} from (8). If he obtains that in a particular region, s is far greater than s_{dummy}, he can conclude that in the region there are a large amount of real applications. If the interests of most applications happen to be the same, he can associate the interest with the region. In this paper, we call this kind of attack regional statistical attack (RSA).

Below, we will illustrate the attack by experiment.

Fig. 7 depicts LBS applications in a particular district at 5:30 p.m. Every point in the figure denotes a real LBS application in the location or a dummy. The grid where more points gathered is a company. The employees of the company stop working at 5:30 and they apply LBS service for bus information at that time. Each time a real application is sent, 19 dummies will be generated. From the figure, we can see that the LBS applications are mainly concentrated in the region. The attacker can analyze the historical data. If in the historical data, at 5:30 p.m. LBS applications about bus information are often sent from the region and the attacker obtains by social engineering that in the region there is a company, he can easily conclude that the company stop working at 5:30 p.m.

3. Enhanced *K*-anonymity method

In this section, we propose methods to defend against LSA and RSA.

3.1. Method to defend against LSA

Since user's historical data can be used for LSA by attackers, we propose two methods to resist the attack.

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

3.1.1. MNAME

In this method, a user stores several user names. Every time LBS service is applied, the user selects one name from the user name set as current user name and sends it to LBS server. In the process of generating these names, uncommon user name should be avoided. The advantage of this method is that privacy protection does not depend on the third party. Since the LBS applications from a user are sporadic, it is hard for an attacker to link two user names to one user. Therefore, mix zone is not needed. This method can preserve privacy efficiently combined with the dummy selection method in [5].

3.1.2. SNAME

In this method, every time a user applies LBS service, the query is changed by the anonymizer and sent to LBS server by the anonymizer. The anonymizer changes every user name to the same one and then sends the query to LBS server. Assume the number of users of the anonymizer is m. As long as m is big enough, LSA can be defended.

3.2. Method to defend against RSA

We propose a method to defend against RSA.

We divide the regions in the map into 4 PLs according to the privacy requirement of the grid.

The first level region mainly includes some secret departments. In this kind of region, the leakage of privacy may infringe on the interest of the department.

The second level region mainly includes the region which can disclose user's personal privacy, such as someone's residence etc. From the LBS applications in this kind of region, the attacker can easily confirm the user's identity and the information related to the region.

The third level region mainly includes the region where there are less people. In this kind of region, privacy is easily violated.

The fourth level region mainly includes the region where there are more people, such as business zone etc. In this kind of region, the possibility of privacy leakage is less.

For the third level region, we will illustrate the problem by an example. In a remote factory, there are two employees. One employee is the attacker and he can obtain the LBS historical data of the region. Although the en-DLS in [5] has less probability in selecting this region as dummy of others, if LBS applications are often sent from the region and the interest of the application is the same, the attacker can conclude that these LBS applications are from the other employee and he can obtain the interest type of the applications. Therefore, in the region where there are less people, the probability of privacy leakage is bigger than the region with more people.

On the contrary, for the fourth level, because in the region, there are more LBS users, the real identities and locations will be protected well by the LBS users who apply LBS service at the same time. Therefore, in the region where there are often a large number of LBS users, the probability of privacy leakage is less.

For the above four level region, based on en-DLS, we propose method to resist RSA. In the $k-1$ dummies, we delete some dummies according to a certain proportion. We make the first level region has a high probability to be a dummy, whereas the fourth level region has a low probability. The advantage is that there will be a lot of LBS queries from the region of high privacy requirement. Then, the attacker cannot distinguish which one is the real application. The rest of dummies are selected by en-DLS.

The algorithm is as follows.

In Algorithm 1, we select $k-1$ dummy locations according to en-DLS firstly. Secondly, in these $k-1$ dummies, we delete some dummies as a proportion of $1/\lambda$. Then we construct regional anonymous set M. First, we select w_1 regions at random from PL 1 regions and put them in M. Second, we select w_2 regions at random from PL 2 regions and put them in M. Third, we select w_3 regions at random from PL 3 regions and put them in M. Fourth, we select w_4 regions at random from PL 4 regions and put them in M. A typical proportion is $\lambda = 4$ and w1:w2:w3:w4 = 8:6:4:1. Then, we choose $(k-1)/\lambda$ dummies from M and add them into set R. Next, we choose a name at random from user name set U and combine the user name with the dummies. At last the k locations will be sent to LBS server. In this algorithm, MNAME is adopted to resist LSA.

Algorithm 1: Statistical Attacks Resilient Algorithm

Input: *k-anonymity k*, PL *i*, proportion λ, collection of user name U, temporal set M
Output: an optimal set of dummy locations R
1 Select dummies as en-DLS and output is R;
2 Delete $(k-1)/\lambda$ dummies from R;
3 Choose w_1 regions from PL 1 regions at random and add them into M;
4 Choose w_2 regions from PL 2 regions at random and add them into M;
5 Choose w_3 regions from PL 3 regions at random and add them into M;
6 Choose w_4 regions from PL 4 regions at random and add them into M;
7 Choose $(k-1)/\lambda$ dummies from M at random and add them into R;
8 Choose a user name in U at random and use it as user name in all locations in R;
9 Output R;

4. Analytic results

4.1. Security analysis

In our methods, the attack to the communication between the user and the LBS server, such as eavesdropping, can be defended by encryption. Next, we mainly focus on the attack in which the LBS server is the attacker. He hopes to obtain some contents about privacy by using historical data.

Entropy is an effective metric to measure the ability of privacy protection. In this section, we use entropy to measure the privacy preserving ability of the proposed method.

First, we analyze the method which uses MNAME in resisting LSA. Assume every user in the system stores m user names. User S also stores m user names. In every LBS query, the probability that a user name is used by the user is p_s.

$$\sum_{s=1}^{m} p_s = 1. \tag{9}$$

Assume that S uses user name u to send a LBS query and there are r users in the same LBS service who have u in their user name set except S. Assume the frequency of LBS application from these r users is q. Assume the time interval between the LBS application from S and next application from S is t. Then in t, every user who has user name u has applied LBS service for $q \times t$ times. Among this applications, the times of applications which use u as the user name is $p_s \times q \times t$. For every application, $k-1$ dummies will be generated. Then in t, u is used as the user name of an application for $k \times r \times p_s \times q \times t + k$ times (the applications from S is included). Among them, the real application from S is 1 times. That is to say, between the two applications, the entropy which the attacker

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

distinguishes real location of S from the anonymous set in time period of t is:

$$H = -\sum_{i=1}^{k\times r\times p_s\times q\times t+k} p_i \cdot \log_2 p_i. \tag{10}$$

If a user does not use uncommon user name, r will be big. The higher is the frequency of user's application, the bigger is q and the real location will be protected against LSA efficiently.

Below, we will discuss the problem in using SNAME to defend against LSA. Assume S is a user who applies LBS service. Assume there are r users using the same anonymizer and the frequency of application of these r users is q. Assume the time interval between the LBS application from S and next application from S is t. Then, in the time interval of t, every user using the same anonymizer applies LBS service for $q \times t$ times. The user name of these users are same as S which is u and in every application, $k-1$ dummy locations will be generated. Then, in t, the number of applications using u as the user name is $k \times r \times q \times t + k$ (S's application is included). Among them, the real application from S is only 1 times. That is to say, between the two applications, the entropy which the attacker distinguishes real location of S from the anonymous set in time period of t is:

$$H = -\sum_{i=1}^{k\times r\times q\times t+k} p_i \cdot \log_2 p_i. \tag{11}$$

At last, we will discuss the security of the method to defend against RSA according to Algorithm 1. Assume in grid A, only S applies LBS service and the time interval between the application from S and next application from S is t. Assume in t, r users apply LBS service. For each user's application, $k-1$ dummies are generated. Among the $k-1$ dummies, $k-k/\lambda$ dummies are generated according to en-DLS. The remaining k/λ dummies are used to preserve the privacy of high PL regions. Let $w = w_1+w_2+w_3+w_4$ and assume the PL of A is i. The number of PL i regions in the map is v_i. According to Algorithm 1, w_i regions will be chosen from v_i at random and be added into M. Thus, the probability of adding A into M, p_{SM} is:

$$p_{SM} = \frac{w_i}{v_i}. \tag{12}$$

Let $g = k/\lambda$. In M, the probability that A is chosen to be a dummy location, p_{SD} is:

$$p_{SD} = \frac{C_{w-1}^{g-1}}{C_w^g} = \frac{g}{w} = \frac{k}{\lambda w}. \tag{13}$$

According to these, we can conclude in one application, the probability that A is chosen to be a dummy location of other applications, p_{SR} is:

$$p_{SR} = p_{SM} \cdot p_{SD} = \frac{kw_i}{\lambda v_i(w_1+w_2+w_3+w_4)}. \tag{14}$$

Between the two applications of S, there are r users applying LBS service. Then, the times that region A becomes dummies of other applications, s_{dummy} is:

$$s_{dummy} = \frac{rkw_i}{\lambda v_i(w_1+w_2+w_3+w_4)} \tag{15}$$

$$H = -p_{real} \cdot \log_2 p_{real} - p_{dummy} \cdot \log_2 p_{dummy}. \tag{16}$$

According to the analysis in Section 2, we can know for LBS server, in the applications from region A, $s = s_{real} + s_{dummy}$. The privacy in the region depends on s_{dummy}. Therefore, we can preserve privacy by ensuring that s_{dummy} is more than s_{real}. In formula (15), s_{dummy} is inversely proportional to v_i and proportional to r. The bigger is w_i, the bigger is s_{dummy}. In practice, we can protect the privacy of the regions by adjusting w_i and v_i. v_i should be small and w_i should be big. That is to say, the regions with a high PL will be given more protection, such as some secret departments and important residence of people. On the other hand, with the popularity of LBS service, the frequency of LBS application will be higher and higher. Thus, a big r can be ensured. Therefore, the method proposed in this paper can solve the problem of RSA along with the popularization of LBS.

In Table 1, we compare the entropy of the proposed method to the method which chooses dummy at random and the methods in [5,13].

In Table 1, only the method in this paper considers LSA and RSA. Therefore, from the point of view of statistics, the entropy of the proposed method in this paper is bigger than $\log_2 k$. The entropy of the method which chooses dummy at random is less than $\log_2 k$ since the method has vulnerabilities in choosing dummies. For CirDummy and GridDummy, when side information is considered, the entropy is lower than $\log_2 k$. For en-DLS, when statistical attacks are not considered, the entropy is $\log_2 k$, whereas when statistical attacks are considered, the entropy is less than $\log_2 k$.

4.2. Performance analysis

In the following, we will analyze the performance of the proposed methods and indicate that they have strong practicality. We compare the performance of the proposed methods with other methods.

4.2.1. Utility

In our methods, for every LBS query, real location is sent with dummies. In MNAME and SNAME, only the user's identity is changed. The accuracy of the location has not been affected. Therefore, the reported locations can still give reasonable query answers to the user. In Algorithm 1, only the dummy locations are changed. The accuracy of the real location has not been affected. So, the LBS server will still give reasonable answers. In short, the utility of the proposed methods is still reasonable as en-DLS.

4.2.2. Communication overhead

Compared with other schemes, MNAME proposed in this paper will not bring extra communication overhead. The LBS application sent from the user includes a real location and $k-1$ dummies and the communication overhead is $O(k)$. The communication overhead of the information sent from LBS server is related to the returned POIs and k. Assume the return POIs are m. Then the communication overhead of LBS server is $O(k^a \times m^b)$, where a and b are constant.

In SNAME, an anonymizer is needed. The communication overhead is divided into the overhead from the user to the anonymizer, the overhead from the anonymizer to LBS server, the overhead from LBS server to the anonymizer, and the overhead from the anonymizer to the user. In them, the communication overhead from the user to the anonymizer only includes the encrypted real user name, location, interest, etc. So, the overhead is a constant. The communication overhead from the anonymizer to LBS server includes the real location and $k-1$ dummies with user name changed. The overhead is $O(k)$. The communication overhead from LBS server to the anonymizer is related to the returned POIs and k. It is $O(k^a \times m^b)$, where a and b are constant. The communication overhead from the anonymizer to the user only includes the POIs for the real application. So it is $O(m)$.

In the method to defend against RSA, some dummies occupy the position of original dummies to provide privacy protection for high PL regions. Therefore no extra communication is needed. The main communication overhead is from distributing the PL of the

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

Table 1
Comparison of security of the methods.

	Proposed method	Random	CirDummy	GridDummy	en-DLS
Entropy	$> \log_2 k$	$< \log_2 k$	$< \log_2 k$	$< \log_2 k$	$\log_2 k$
Side information	Y	N	N	N	Y
Resist LSA	Y	N	N	N	N
Resist RSA	Y	N	N	N	N

regions in the map. The information of PL is not often updated. It may be updated once a month or once a year. So, this part of communication overhead can be ignored in theoretical analysis.

In Table 2, we compare the communication overheads of the proposed methods to the method which chooses dummy at random and en-DLS. In the table, C denotes a constant.

4.2.3. Computational cost

The computational cost mainly includes the encryption and decryption of locations.

In MNAME, a user chooses a user name at random and generates dummies. To prevent the attacker from eavesdropping, the transmitted information should be encrypted and the results will be sent to the LBS server. The computational cost of the user is $O(k)$. After the information arrives at LBS server, the server encrypts the returned POIs and sends to the user. The computational cost of LBS server is $O(k^a \times m^b)$, where a and b are constant. The user obtains the encrypted POIs. Then he decrypts the POIs. The computational cost is $O(k^a \times m^b)$.

In SNAME, when a LBS query is sent, the user's computational cost includes the encryption of real user identity and real location. The cost is $O(C)$. The anonymizer generates $k-1$ dummy locations and sends the results to LBS server. The computational cost is $O(k)$. After LBS server obtains the application, POIs are generated. Then the POIs are encrypted and sent to the anonymizer. The computational cost of LBS server is $O(k^a \times m^b)$. The anonymizer decrypts the POIs. Then the anonymizer encrypts the results and sends them to the user. The computational cost is $O(k^a \times m^b + m)$. After the user receives the m encrypted POIs, he decrypted them and gets the last results. The computational cost is $O(m)$. Therefore, the total computational cost of the user is $O(C + m)$. The total cost of the anonymizer is $O(k^a \times m^b + m + k)$.

In the method to defend against RSA, the original dummies are only substituted by the dummies selected according to the PL. Therefore, only the computation of selecting dummies is added. Since the computation cost of selecting dummies is very small, this computation cost is ignored in performance analysis.

In Table 3, the computational costs of the methods are compared.

4.2.4. Storage overhead

In MNAME, for mobile devices, the storage of user name set is added. The storage overhead is related to number of the user name stored. It is $O(n)$, where n is the number of the user name stored. In SNAME, because an anonymizer is used, extra storage overhead is not needed for the mobile user. The storage overhead of the anonymizer is related to the number of the users to apply LBS service. In the method to defend against RSA, users need to obtain and store the PLs in the map. The storage overhead is $O(r \times r)$. Since the PLs will not often be updated, the storage overhead is a constant.

5. Experiments and results

Below, we will verify the performance of the proposed methods by experiment. The experiment focuses on the ability of privacy protection of the methods, user's communication overhead, and user's computational cost.

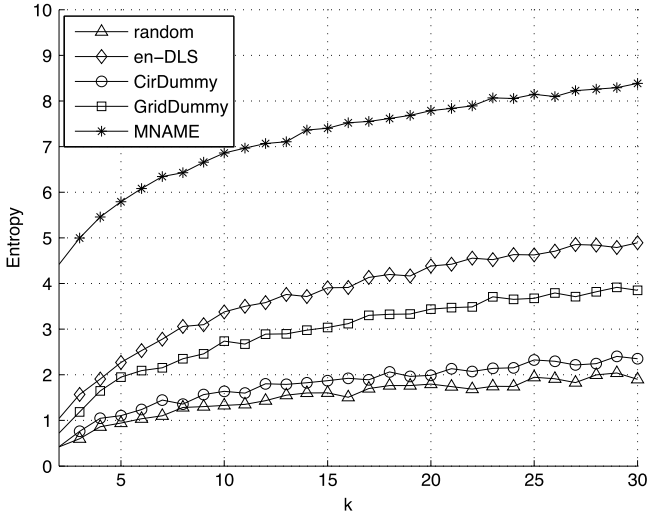

Fig. 8. Entropy vs. k.

5.1. Experimental settings

The machine which we do our experiment is a computer with Intel Pentium CPU G630 2.7 GHz and 8.0 GB RAM. The operation system of the computer is Win7 64bit. Our experiment is under Matlab 2012b. In our experiment, the map is divided into grids of 1000×1000. 10 000 users are randomly distributed into the grids. The POIs in the map are totally 1000.

5.2. Simulation results

5.2.1. Entropy

In MNAME, there are 1000 user names for all users. Assume that every user chooses 5 names as user name set from the user names. The frequency of user's LBS application is 1 per hour. In Fig. 8, we compare the entropy of our method to the others which use dummy in preserving privacy.

In Fig. 8, en-DLS is the method which is proposed in [5]. Random is the method that selects dummies at random. CirDummy and GridDummy are the virtual circle and virtual grid proposed in [13]. MNAME is the method proposed in this paper. From Fig. 8, we can see the entropy of MNAME is the biggest. It is bigger than $\log_2 k$. The reason is that MNAME considers LSA and uses multiple user names. The LBS queries from others which using the same user name protect the user's real query.

In the figure, only MNAME is investigated. The entropy of SNAME is related to the users in the same anonymizer. It is omitted.

Next, we will investigate the entropy that the attacker distinguishes real application from a particular region. Assume the PL of the region is 1. $\lambda = 4$ and $w_1 : w_2 : w_3 : w_4 = 8 : 6 : 4 : 1$. In the map, there are totally 100 regions whose PL is 1.

In Fig. 9, random is the method which selects dummies at random. REGP is the method to defend against RSA which is proposed in this paper. From the figure, we can see the method proposed in this paper has a bigger entropy than the method that selects dummies at random. The entropy of the random method is about 24.9% of REGP.

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

Table 2
Comparison of communication overhead of the methods.

	MNAME	SNAME	Random	en-DLS
From user to LBS	$O(k)$	–	$O(k)$	$O(k)$
From LBS to user	$O(k^a \times m^b)$	–	$O(k^a \times m^b)$	$O(k^a \times m^b)$
From user to anonymizer	–	$O(C)$	–	–
From anonymizer to LBS	–	$O(k)$	–	–
From LBS to anonymizer	–	$O(k^a \times m^b)$	–	–
From anonymizer to user	–	$O(m)$	–	–

Table 3
Comparison of computational cost of the methods.

	MNAME	SNAME	Random	en-DLS
User	$O(k^a \times m^b + k)$	$O(C + m)$	$O(k^a \times m^b + k)$	$O(k^a \times m^b + k)$
LBS server	$O(k^a \times m^b)$	$O(k^a \times m^b)$	$O(k^a \times m^b)$	$O(k^a \times m^b)$
Anonymizer	–	$O(k^a \times m^b + m + k)$	–	–

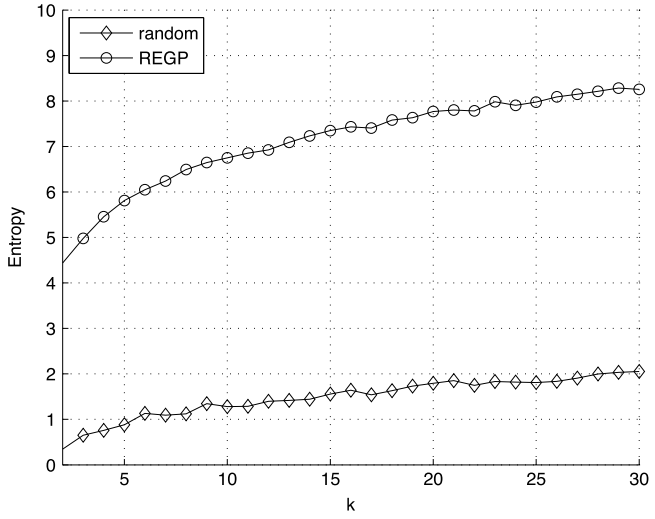

Fig. 9. Entropy vs. k for regions.

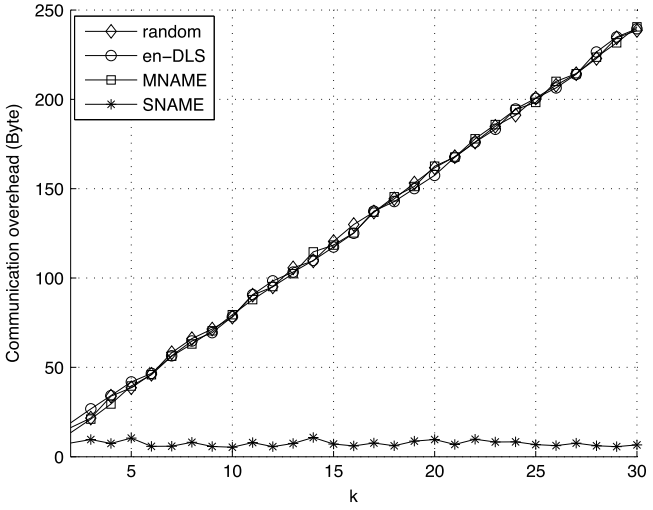

Fig. 10. Communication overhead of users.

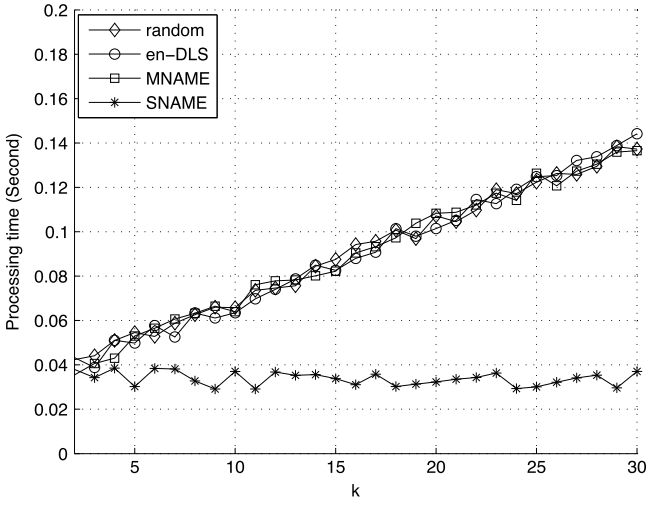

Fig. 11. Computational cost of users.

5.2.2. Communication overhead

Fig. 10 describes the relationship between k and communication overheads of users. The overhead of sending a 2D location is about 8 bytes. In the figure, random is the method that selects dummies at random. MNAME and SNAME are the methods proposed in this paper. In the figure, the communication overheads of random, en-DLS, and MNAME are similar. In random, en-DLS, and MNAME, dummies are generated in the user side. Therefore the communication overhead increases with the increase of k. In SNAME, users only need to send real location to the anonymizer. So, the communication overhead is a constant.

5.2.3. Computational cost

In Fig. 11, the computational costs of users mainly include the encryption and decryption of locations and returned POIs. In the figure, the computational costs of random, en-DLS, and MNAME are similar. Since in these three methods, dummies are generated at the user side and information is encrypted and decrypted at the user side, the computational cost is higher than SNAME. The computational cost of random, en-DLS, and MNAME increases with the increase of k. In SNAME, anonymizer is adopted to preserve privacy. Therefore, the computational cost only includes the cost of encryption and decryption of real location and returned POIs.

6. Related work

In this section, we review some researches on user's privacy preservation for LBSs.

It is of great significance to ensure user's privacy and security when delivering high-precision location services to users. Many solutions have been documented in the literature. These solutions can be divided into two categories according to the technology used. They are obfuscation and anonymity [14]. In applying LBS service, obfuscation does not send real location to the LBS server, but conceals the real location [15–18]. This kind of methods mainly include adding noise and sending the similar location to the

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017

LBS server. In [14], the author puts forward to collect locations and interests from the encounter LBS users and to conceal the real location by using historical data of different users. Although this kind of methods can preserve privacy in some degree, the precision of LBS service is not high due to the locations sent are not accurate. Different from obfuscation, anonymity methods [4,19,20] are designed to make the user's real location cannot be distinguished from the sent locations by LBS server, so as to protect the real location. The main technologies used in anonymity methods include pseudo name, *k-anonymity* and others. In [4], the author proposes mix zone. In mix zone, users are protected by pseudo name. But this method lacks practicality and it is hard to be applied in reality. On the other hand, *k-anonymity* is the main method in the field of privacy preservation currently. It is first used in protecting the data in database [19]. It is used in protecting location privacy by Gruteser and Grunwald [8]. In order to achieve *k-anonymity*, there are two main methods: Cloaking and Using Dummy. In Cloaking, there is an anonymizer. The anonymizer extends the real location into a big Cloaking area to ensure that in the area there are at least k users sending LBS query at the same time. Then the anonymizer sends the whole Cloaking area to LBS server. However, in [21], the author points out the weakness in Cloaking. Another method to achieve *k-anonymity* is to send the real location with $k-1$ dummies so as to make the LBS server cannot distinguish which one is the real location. Ideally, the probability of every location to be the real location is $1/k$. In [22], the author studies the trilateral trade-off between privacy, service quality, and bandwidth. The author points out dummy-based location privacy-preserving mechanisms offer the best protection for a given combination of quality and bandwidth constraints. In [23], Kido et al. realize *k-anonymity* using dummies. They generate the dummy locations by random moving. The author in [13] designs two schemes for selecting dummies. They are CirDummy and GridDummy. In CirDummy, dummy locations are generated based on a virtual circle, whereas in GridDummy, dummy locations are generated based on a virtual grid. The dummy locations in these methods are carefully selected to ensure that the Cloaking area is big enough. In [5], the author notices that the effect of side information on user's privacy disclosure, especially for the terrain constraints in the city. The author proposed Dummy-Location Selection (DLS) scheme. When a user sends a LBS query, dummies are selected in the locations that have similar query probability as the real location. In order to increase coverage area of DLS, en-DLS is proposed. In en-DLS, the cloaking area is big enough at the same time big entropy is achieved.

According to the anonymizer is used or not, privacy protection methods at present can be divided into trusted third part (TTP) schemes and trusted third part free (TTP-free) schemes [24]. In TTP schemes, user's privacy protection is realized by a trusted third party which is called anonymizer. The anonymizer receives application from the user and protects the application by using Cloaking or other methods. Then the protected application will be send to the LBS server by the anonymizer. The disadvantage is the anonymizer can be the bottleneck of the system and it can be the main target of attacks. Once the anonymizer is compromised, all the mobile users using the anonymizer may be faced with security threats, namely single point of failure. Privacy preserving schemes using anonymizer include the methods in [24–28]. Schemes which do not use anonymizer are known as TTP-free schemes. In these schemes, the privacy preservation of mobile devices does not depend on the trusted third party. Users generate privacy preserving information rely on the smart mobile devices on their own and the results will be sent to LBS server directly. This kind of methods mainly includes the methods in [5,9,10,23,29,30].

In addition, according to the frequency of a user request to LBS server, the problem of privacy protection can be divided into sporadic privacy protection and continuous privacy protection. In sporadic privacy protection, the period between two LBS queries from a user cannot be neglected. For example, a user sends LBS query to obtain the restaurant address nearby. The next LBS query may be applied a long period after current query. In continuous privacy protection, users send LBS queries continuously. For example, moving cars on the road send continuous LBS queries to LBS server to obtain current traffic information. In this kind of LBS service, the trajectory of users can also reveal users' privacy. Therefore, in this kind of privacy protection, not only user's location should be preserved, but also the trajectory needs to be protected. Schemes of continuous privacy protection mainly include the schemes in [18,31–34].

7. Conclusion

In this paper, we first presented the vulnerabilities existing in en-DLS. The attacker can violate a user's privacy by LSA and RSA using historical data. Then, we gave out methods to defend against these two attacks. For LSA, we proposed MNAME and SNAME. For RSA, we proposed to divide the regions in the map into different PLs. We replaced some dummies with the locations of regions according to the PLs. Furthermore, we analyzed the privacy preserving ability of the proposed methods by using entropy. In the analysis, we gave out the relation in the entropy, the frequency of LBS application, and the duration of historical data obtained by the attacker. The results of simulation showed that the methods proposed in this paper protect user's privacy against historical statisticardagna2007locationardagna2007locationduckham2005formal attack with high performance. The future work mainly focuses on the randomness of user's application and the drastic changing of LBS applications in different time in a day.

Acknowledgment

The authors would like to extend their sincere appreciation to the Deanship of Scientific Research at King Saud University for its funding of this research through the Research Group Project no. RGP-281.

References

[1] Y. Zhang, D. Zhang, M.M. Hassan, A. Alamri, L. Peng, Cadre: Cloud-assisted drug recommendation service for online pharmacies, Mob. Netw. Appl. 20 (3) (2015) 348–355.
[2] J.H. Reed, K.J. Krizman, B.D. Woerner, T.S. Rappaport, An overview of the challenges and progress in meeting the e-911 requirement for location service, IEEE Commun. Mag. 36 (4) (1998) 30–37.
[3] Q. Liu, Y. Ma, M. Alhussein, Y. Zhang, L. Peng, Green data center with iot sensing and cloud-assisted smart temperature control system, Comput. Netw. 101 (2016) 104–112.
[4] A.R. Beresford, F. Stajano, Location privacy in pervasive computing, IEEE Pervasive Comput. 2 (1) (2003) 46–55.
[5] B. Niu, Q. Li, X. Zhu, G. Cao, H. Li, Achieving *k*-anonymity in privacy-aware location-based services, in: INFOCOM, 2014 Proceedings IEEE, IEEE, 2014, pp. 754–762.
[6] B. Niu, Q. Li, X. Zhu, G. Cao, H. Li, Enhancing privacy through caching in location-based services, in: 2015 IEEE Conference on Computer Communications, (INFOCOM), IEEE, 2015, pp. 1017–1025.
[7] R. Shokri, J. Freudiger, M. Jadliwala, J.-P. Hubaux, A distortion-based metric for location privacy, in: Proceedings of the 8th ACM Workshop on Privacy in the Electronic Society, ACM, 2009, pp. 21–30.
[8] M. Gruteser, 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.
[9] B. Niu, X. Zhu, H. Chi, H. Li, 3plus: Privacy-preserving pseudo-location updating system in location-based services, in: Wireless Communications and Networking Conference (WCNC), 2013 IEEE, IEEE, 2013, pp. 4564–4569.
[10] X. Zhu, H. Chi, B. Niu, W. Zhang, Z. Li, H. Li, Mobicache: When *k*-anonymity meets cache, in: Global Communications Conference (GLOBECOM), 2013 IEEE, IEEE, 2013, pp. 820–825.
[11] A.R. Beresford, F. Stajano, Mix zones: User privacy in location-aware services, 2004, pp. 127–131.
[12] Y. Zhang, M. Qiu, C.W. Tsai, M.M. Hassan, A. Alamri, Health-cps: Healthcare cyber-physical system assisted by cloud and big data, IEEE Syst. J. PP (99) (2015) 1–8. http://dx.doi.org/10.1109/JSYST.2015.2460747.

[13] H. Lu, C.S. Jensen, M.L. Yiu, Pad: privacy-area aware, dummy-based location privacy in mobile services, in: Proceedings of the Seventh ACM International Workshop on Data Engineering for Wireless and Mobile Access, ACM, 2008, pp. 16–23.
[14] B. Niu, X. Zhu, X. Lei, W. Zhang, H. Li, Eps: Encounter-based privacy-preserving scheme for location-based services, in: Global Communications Conference (GLOBECOM), 2013 IEEE, IEEE, 2013, pp. 2139–2144.
[15] M. Duckham, L. Kulik, A formal model of obfuscation and negotiation for location privacy, in: Pervasive Computing, Springer, 2005, pp. 152–170.
[16] D. Quercia, I. Leontiadis, L. McNamara, C. Mascolo, J. Crowcroft, Spotme if you can: Randomized responses for location obfuscation on mobile phones, in: 2011 31st International Conference on Distributed Computing Systems, (ICDCS), IEEE, 2011, pp. 363–372.
[17] C.A. Ardagna, M. Cremonini, E. Damiani, S.D.C. Di Vimercati, P. Samarati, Location privacy protection through obfuscation-based techniques, in: Data and Applications Security XXI, Springer, 2007, pp. 47–60.
[18] R.-H. Hwang, Y.-L. Hsueh, H.-W. Chung, A novel time-obfuscated algorithm for trajectory privacy protection, IEEE Trans. Serv. Comput. 7 (2) (2014) 126–139.
[19] L. Sweeney, *k*-anonymity: A model for protecting privacy, Internat. J. Uncertain. Fuzziness Knowledge-Based Systems 10 (05) (2002) 557–570.
[20] A. Machanavajjhala, D. Kifer, J. Gehrke, M. Venkitasubramaniam, l-diversity: Privacy beyond *k*-anonymity, ACM Trans. Knowl. Discov. Data (TKDD) 1 (1) (2007) 3.
[21] R. Shokri, C. Troncoso, C. Diaz, J. Freudiger, J.-P. Hubaux, Unraveling an old cloak: *k*-anonymity for location privacy, in: Proceedings of the 9th Annual ACM Workshop on Privacy in the Electronic Society, ACM, 2010, pp. 115–118.
[22] M. Herrmann, C. Troncoso, C. Diaz, B. Preneel, Optimal sporadic location privacy preserving systems in presence of bandwidth constraints, in: Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society, ACM, 2013, pp. 167–178.
[23] H. Kido, Y. Yanagisawa, T. Satoh, An anonymous communication technique using dummies for location-based services, in: International Conference on Pervasive Services, 2005. ICPS'05. Proceedings, IEEE, 2005, pp. 88–97.
[24] T. Peng, Q. Liu, G. Wang, Enhanced location privacy preserving scheme in location-based services, IEEE Syst. J. PP (99) (2014) 1–12.
[25] K. Vu, R. Zheng, J. Gao, Efficient algorithms for *k*-anonymous location privacy in participatory sensing, in: INFOCOM, 2012 Proceedings IEEE, IEEE, 2012, pp. 2399–2407.
[26] G. Ghinita, K. Zhao, D. Papadias, P. Kalnis, A reciprocal framework for spatial *k*-anonymity, Inf. Syst. 35 (3) (2010) 299–314.
[27] M.F. Mokbel, C.-Y. Chow, W.G. Aref, The new casper: query processing for location services without compromising privacy, in: Proceedings of the 32nd International Conference on Very Large Data Bases, VLDB Endowment, 2006, pp. 763–774.
[28] C.-Y. Chow, M.F. Mokbel, W.G. Aref, Casper*: Query processing for location services without compromising privacy, ACM Trans. Database Syst. (TODS) 34 (4) (2009) 24.
[29] M.L. Yiu, C.S. Jensen, X. Huang, H. Lu, Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services, in: IEEE 24th International Conference on Data Engineering, 2008, ICDE 2008, IEEE, 2008, pp. 366–375.
[30] X. Liu, K. Liu, L. Guo, X. Li, Y. Fang, A game-theoretic approach for achieving *k*-anonymity in location based services, in: INFOCOM, 2013 Proceedings IEEE, IEEE, 2013, pp. 2985–2993.
[31] C.-Y. Chow, M.F. Mokbel, Trajectory privacy in location-based services and data publication, ACM SIGKDD Explor. Newslett. 13 (1) (2011) 19–29.
[32] J. Krumm, Realistic driving trips for location privacy, in: Pervasive Computing, Springer, 2009, pp. 25–41.
[33] B. Hoh, M. Gruteser, Protecting location privacy through path confusion, in: First International Conference on Security and Privacy for Emerging Areas in Communications Networks, 2005, SecureComm 2005, IEEE, 2005, pp. 194–205.
[34] T. Xu, Y. Cai, Exploring historical location data for anonymity preservation in location-based services, in: INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, IEEE, 2008, pp. 547–555.

Yanming Sun received the M.S. degree in computer application from Northeastern University, Shenyang, China, in 2005 and the Ph.D. degree from Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2013, respectively. Since December 2013, he has been with Huazhong University of Science and Technology, Wuhan, China, as a Postdoctoral Fellow. His current research interests include network security, vehicular ad hoc network security, privacy for wireless networks, etc.

Min Chen is a professor in School of Computer Science and Technology at Huazhong University of Science and Technology (HUST). He is Chair of IEEE Computer Society (CS) Special Technical Communities (STC) on Big Data. He was an assistant professor in School of Computer Science and Engineering at Seoul National University (SNU) from Sep. 2009 to Feb. 2012. He worked as a Post-Doctoral Fellow in Department of Electrical and Computer Engineering at University of British Columbia (UBC) for three years. Before joining UBC, he was a Post-Doctoral Fellow at SNU for one and half years. He received Best Paper Award from IEEE ICC 2012, QShine 2008, and IndustrialIoT 2016. He serves as editor or associate editor for Information Sciences, Wireless Communications and Mobile Computing, IET Communications, IET Networks, Wiley I. J. of Security and Communication Networks, Journal of Internet Technology, KSII Trans. Internet and Information Systems, International Journal of Sensor Networks. He is managing editor for IJAACS and IJART. He is a Guest Editor for IEEE Network, IEEE Wireless Communications Magazine, etc. He is Co-Chair of IEEE ICC 2012-Communications Theory Symposium, and Co-Chair of IEEE ICC 2013-Wireless Networks Symposium. He is General Co-Chair for the 12th IEEE International Conference on Computer and Information Technology (IEEE CIT-2012) and Mobimedia 2015. He is General Vice Chair foTridentcom 2014. He is Keynote Speaker for CyberC 2012, Mobiquitous 2012 and Cloudcomp 2015. He has more than 260 paper publications, including 120+ SCI papers, 50+ IEEE Trans./Journal papers, 6 ISI highly cited papers and 1 hot paper. He has published two books: OPNET IoT Simulation (2015) and Big Data Inspiration (2015) with HUST Press, and a book on big data: Big Data Related Technologies (2014) with Springer Series in Computer Science. His Google Scholars Citations reached 5700+ with an h-index of 36. His top paper was cited 690 times, while his top book was cited 420 times as of Aug. 2015. He is an IEEE Senior Member since 2009. His research focuses on Internet of Things, Mobile Cloud, Body Area Networks, Emotion-aware Computing, Healthcare Big Data, Cyber Physical Systems, and Robotics, etc.

Long Hu is a visiting student at the Department of Electrical and Computer Engineering, University of British Columbia. He is a Ph.D. candidate in School of Computer Science and Technology, Huazhong University of Science and Technology (HUST), China, since 2014. His research includes the Internet of Things, machine-to-machine communications, body area networks, body sensor networks, RFID, e-healthcare, and mobile cloud computing.

Yongfeng Qian received the M.S. degree in school of mathematics and statistics from Huazhong University of Science and Technology, Wuhan, China in 2015. Currently, she is a Ph.D. candidate in School of Computer Science and Technology, HUST since 2015. Her research interests include Big data, open source software, server virtualization, cloud computing and the Internet of Things.

Mohammad Mehedi Hassan is currently an Assistant Professor of Information Systems Department in the College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Kingdom of Saudi Arabia. He received his Ph.D. degree in Computer Engineering from Kyung Hee University, South Korea in February 2011. He has published over 100+ research papers in the journals and conferences of international repute. He has served as, chair, and Technical Program Committee member in numerous international conferences/workshops like IEEE HPCC, ACM BodyNets, IEEE ICME, IEEE ScalCom, ACM Multimedia, ICA3PP, IEEE ICC, TPMC, IDCS, etc. He has also played role of the guest editor of several international ISI-indexed journals. He received Best Paper Award from CloudComp'14 conference at China in 2014. He received Excellence in Research Award from CCIS, KSU in 2014 & 2015. His research areas of interest are cloud federation, multimedia cloud, sensor-cloud, Internet of Things, Big data, mobile cloud, cloud security, IPTV, sensor network, 5G network, social network, publish/subscribe system and recommender system.

Please cite this article in press as: Y. Sun, et al., ASA: Against statistical attacks for privacy-aware users in Location Based Service, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.06.017