Another Dummy Generation Technique in Location-Based Services

Hyo Jin Do, Young-Seob Jeong, Ho-Jin Choi, and Kwangjo Kim
School of Computing
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
{hjdo, pinode, hojinc, kkj}@kaist.ac.kr

***Abstract*—With the proliferation of mobile devices, many users now take advantage of location-based services that use their current position. However, careful consideration should be made when sending one's location to another as the location often includes personal attributes such as home address and reveals private information such as health or religion. To resolve this issue, a dummy generation technique is widely used. This technique protects the location privacy of a user by generating false position data (dummy) along with the true position data to obfuscate an adversary. However, the current dummy generation technique rarely assumes any prior knowledge held by the attacker that may allow them to reduce the level of uncertainty about the true location.**

In this paper, we propose a dummy generation method that is resistant to adversaries who have information about the user as well as external spatiotemporal knowledge. Our method uses conditional probabilities to generate realistic false locations at which the user is highly likely to be located at the given time and add more weight to the vulnerable location and time pairs. We first describe the strategy for the adversary and present our dummy generation method which is simple and effective for preventing the described attack. Experimental results show that our method obfuscates the true location more successfully compared to other approaches.

I. Introduction

Smartphones have become highly portable and accurate sensing devices in recent years. Many researchers have tried to collect data from smartphones in order to utilize the data for numerous context-aware applications. Among the applications currently available, location-based services are among the most widely deployed types and are capable of offering personalized real-time services. For example, Foursquare [1] is a popular location-based social networking application that provides personalized local searches. Shop Alert [2] is a location-aware marketing program in the US that delivers coupons and special deals to consumers when they are near a shop. OnStar [3], provided by General Motors, supports location-aware services such as stolen vehicle tracking, and there are a myriad of other location-based applications [4]–[7] used in various other fields, including weather reporting, medical services, real-time Q & A, recommendation, and advertisement.

When using location-based services, a user sends his/her current position along with requests to a service provider through communication networks, and the service provider offers services based on the location coordinates. The common vulnerability in this scenario is the case in which an adversary is the owner, maintainer, or controller of the service provider. This is critical because servers store personal location data, including home addresses, and they may reveal sensitive information such as health conditions or religion. To counter against this vulnerability, one of the popular methods is to hide the true location from service providers by generating false noise locations called "dummy locations". However, existing dummy generation methods rarely assume that an attacker has prior knowledge about the user and the environment, which may give clues that distinguish the true location from the dummy location.

There has been an increase in the rates of social network service usage by all age groups in recent years. A pertinent privacy concern arises owing to the fact that social networking sites reveal personal information on profile pages or personal posts. We propose an attack scenario in which an adversary collects some user information, for example, their home or office address, from social networking sites and utilizes the information to distinguish their true location from the dummy locations. In addition, we assume that the adversary has spatiotemporal information such as the locations of residences, restaurants, and offices, and the times people normally sleep, eat, work, *etc*. In this attack scenario, it is critical that all generated dummy locations must be convincing to adversaries; yet not much research has proven simple and effective in this particular attack. In this paper, we propose a method that generates dummy locations that are highly convincing in such an attack strategy using a simple statistical approach. To sum up, we define our attack scenario and the goal of our method as follows:

- **Attack Scenario -** An attacker has prior knowledge about a target user and external spatiotemporal information, belonging to a context-linking attack [29]. The attacker tries to distinguish the user's true location from dummy locations using this information.
- **Goal -** We propose a method to generate realistic dummy locations robust against this attack scenario.

Our dummy generation method carefully chooses dummy locations from the top k realistic locations calculated by conditional probabilities. In addition, we pay special attention to locations that may have vulnerability in specific spatiotemporal

contexts by adding a weight factor. Experiments using real data have demonstrated that our method is more resistant to the given attack: the success probability of the attacker in our method is 4.6% lower than the state-of-the-art approach and 35.6% lower than the random approach, on average. Our approach generates more realistic dummy locations according to the time. Moreover, our dummy locations are more probable when real locations with low query probabilities are given. Our dummy generation approach is sufficiently simple to be utilized in real-world applications and obfuscates the true location among the dummy locations more successfully as compared to other methods.

The rest of this paper is organized as follows: After reviewing related work in Section II, we explain our threat model and dummy generation in Section III. We explain our proposed method and present our experimental design and evaluation in Sections IV and V, respectively. Finally, we make a conclusion and discuss future work in Section VI.

II. Related Work

Location privacy is defined as the ability to prevent other parties from learning one's current or past location [8]. Extensive work has been carried out to show the vulnerability of location-based services and to assess the importance of location privacy. For example, Liao *et al.* [9] infer users' activities from the frequency of their visits to certain locations. De Mulder *et al.* [10] identify a mobile user from previous movements. Some research [11]–[13] deduce the home addresses of individuals from observing location traces. Matsuo *et al.* [14] infer personal information from users' indoor location data.

To protect location privacy while using location-based services, many mechanisms have been suggested. The previous research can be largely categorized to three main approaches: spatial or temporal cloaking, using mix zones, and generating dummies. The cloaking approach designs cloaking boxes that each contains k users. A user is concealed in the cloaking box and the box's location information is sent to the service provider [15]–[19]. The problem with this method is that the spatial or temporal accuracy of location information is reduced. Further, the method generally requires third-party server to save global knowledge of a large number of users, which is impossible for real-world applications and can yield a performance bottleneck. The second type of location privacy protection mechanism utilizes a mix zone [20], [21], which is a region where there is no update of location information. The trusted server collects the pseudonyms of users within a mix zone and assigns new pseudonyms to confuse attackers. This method also requires a trusted server for processing the perturbation algorithm and location privacy is degraded when there are too few people in the mix zone. Lastly, the dummy generation approach sends several fake locations, called dummy locations, along with the true location to servers, increasing uncertainty for the adversary. This method is a mobile-based one that does not require any trusted servers and is known to achieve a similar level of privacy without loss of location accuracy. However, there are additional communication costs for the fake locations; thus, several cost reduction techniques have been devised [22]. It is also challenging to create realistic dummy locations, in particular, if an attacker has additional temporal or context information [23] that helps to discover the true location. Very early work was carried out by Kido *et al.* [22], who established dummy trajectories in which the next position is selected from the neighborhood of current positions. The starting points can be selected randomly. Similarly, Krumm [24] fakes a users' driving movements by using a database of actual GPS tracks from 253 drivers. They compute the probability of a given position being a starting or ending point to make a more realistic model. Chow *et al.* [25] generate fake location traces by leveraging Google Maps. They add simulated stops and noises in the routes in Google Maps and output a fraction of the points according to the desired time range.

In this paper, we focus ourselves on the dummy generation approach. Different from the existing approaches, we assume that an attacker has additional context knowledge of the user's location-related information, such as home or office addresses, and external spatiotemporal information, such as addresses of restaurants, *etc.* Few previous works assume prior information held by an attacker. For example, Shokri *et al.* [26] provide a formal framework for the analysis of location-privacy protection mechanisms in which the attacker's prior mobility information about each user is assumed. This information is encoded either in the form of traces or as a matrix of transition counts and then modeled as a Markov chain. Another study [27] proposes a mechanism which preserves the optimal location privacy of a user given the user's service quality constraints against an optimal inference attack. The study assumes that the adversary knows each user's location-based service access patterns and underlying obfuscation algorithm, and formalize the problem as a zero-sum Bayesian Stackelberg game [27]. This research differs from our approach, as they protect location privacy by transforming actual locations into pseudo-locations, whereas we use dummy locations as well as the actual location. Similar to our work, Niu *et al.* [28] propose Dummy Location Selection (DLS) algorithms considering the side information available to adversaries. They assume that the adversary knows the user's query probabilities and generate realistic dummies using an entropy-based scheme. To the best of our knowledge, we establish a novel method that is resistant to another particular assumption of the adversary's prior knowledge, a user's profile and spatiotemporal information. We assume a more realistic scenario that is highly likely to happen at current times when various social networking sites and local search services are widely deployed.

III. Our Threat Model and Dummy Generation

A. Threat Model

A common architecture for location-based services consists of mobile devices, positioning systems, communication network, and service providers as shown in Fig. 1. Sensors in the mobile devices send a user's current location to the service

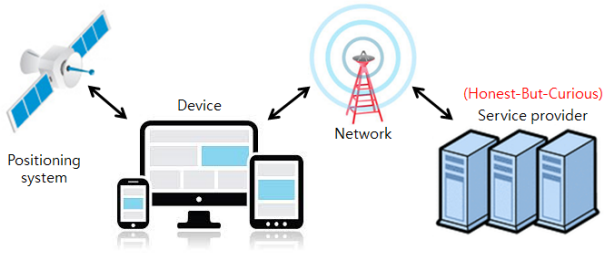

Fig. 1. A common architecture for location-based services.

provider using communication network, which computes and responds to queries based on the user's location coordinates. To make our threat model, we assume an honest-but-curious (HBC) service provider as the adversary. An HBC service provider behaves according to a predefined protocol but tries to derive sensitive information, *i.e.*, the real location in this case, from the stored data. Among various attack scenarios mentioned in [23], we focus ourselves on the single-position attack where the adversary analyzes a single query from the user. Additionally, the user is assumed to reveal his/her location in a sporadic way. In contrast, the device and the positioning system are 100% trustful. We focus ourselves on a device-based dummy generation technique in which the decisions made by the technique are trusted.

B. Dummy Generation

To deal with the threat model described in Section III-A, a dummy generation technique is devised which hides the true location from the service provider by sending one or more false locations, namely dummy locations or dummies, together with the true one. As illustrated in Fig. 2, the overall procedure operates as follows:

1) The user's device is at position A.
2) The user sends position data A, including dummy locations, B, C, D, and E to the service provider.
3) The service provider creates services for all position data, A to E, and sends the services to the user.
4) The user receives all the services and selects only the desired service, *i.e.*, that for A.

The real location A is sent to the service provider as well as the four dummy locations B, C, D, and E (k=5). The user knows the true location, whereas the service provider does not. Thus, the service provider cannot distinguish the true location from a set of k received locations (1 real location and $k-1$ dummy locations). In this way, location privacy is preserved, achieving k-anonymity [29].

C. Our Attack Scenario

We propose an attack scenario in which the HBC service provider performs a context-linking attack [29] which assumes that the adversary has prior knowledge about the user as well as the external spatiotemporal information. Owing to the wide use of social networking sites, homepages, and other

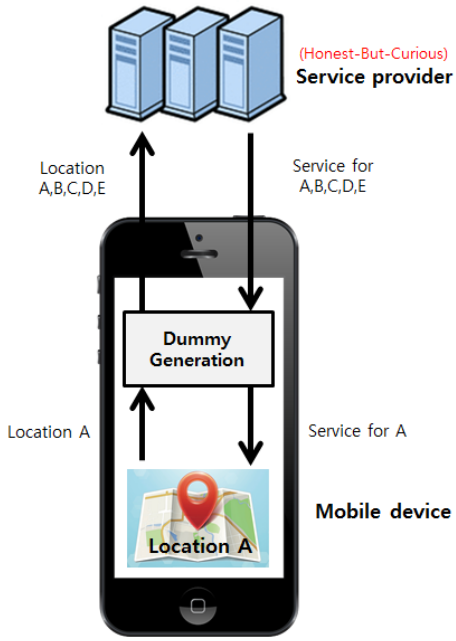

Fig. 2. Operation of a location-based service with the dummy generation.

Algorithm 1 Attacker's Strategy

User information: L (locations related to a user, *e.g.*, home or office address)
Background information: TL (possible time and location pairs, *e.g.*, (12 p.m., restaurant))
Query input: t (time), C (k query locations), R (possible answers)
Output: r (real location guess)
if $(t, c) \in TL$ where $\forall c \in C$ **then**
  $R \leftarrow c$
else
  if $c \in L$ **then**
    $R \leftarrow c$
  else $R \leftarrow C$
  end if
end if
return $r \leftarrow$ random pick from R

vulnerable internet websites, adversaries can easily collect the target user's information, for instance, their home, office, school information, *etc.*, and use the information to discern the true location. For example, assume that a victim is at home and is using a location-based service. The location-based service will send the home address along with some random dummy locations to the HBC service provider. If the HBC service provider also knows the victim's home address, the service provider may guess that the victim is at home since the location is more probable than any unfamiliar locations. Further, the adversary may infer the real location using their spatiotemporal knowledge. For example, if there is a location of a restaurant among the candidates queried at lunchtime,

the adversary may infer that the real location is likely to be the restaurant. This prior knowledge raises a critical issue for existing dummy generation techniques because the knowledge may help to guess the true location from among several candidates.

The detailed procedure of our attack scenario is described in Algorithm 1. First, an attacker guesses the real location from among the possible answers that the adversary thinks are realistic locations given a particular time. If there are no such location and time pairs, the attacker selects user-related locations, and otherwise, random locations. If multiple matches exist, the adversary selects randomly among the matches. In this paper, we generate dummy locations that are robust to the attack scenario presented above.

IV. Our Proposed Method

In this section, we describe a new dummy generation method that is robust to an HBC service provider that has prior knowledge about the user and the context (Section III-C). Our approach is based on the following ideas:

- Generate dummy locations where the target user frequently visits at the given time.
- Generate dummy locations that are predicted to be vulnerable from the user's perspective.

We calculate conditional probabilities to tackle the first idea, and add a weighting scheme to consider the second idea.

A. Generating Frequent Locations

Our approach carefully chooses dummy locations using conditional probability of locations given a time, predicting the behavior of the target user at a particular time of day. Equation (1) calculates the probability that a user is at a specific $location$ at a given $time$.

$$P(location|time) = \frac{P(location \cap time)}{P(time)} \tag{1}$$

Equation (1) calculates the probabilities of joint of events location and time, and the probabilities of times. Additionally, we normalize the data by adding 1 occurrence to every location/time pair to prevent 0 probabilities. After calculating the $P(location|time)$ of all possible locations at the given time, we generate dummy locations that result in the highest probabilities excluding the real location. If the values of $P(location|time)$ are identical, we select the locations based on the value of $P(location)$. In this way, we can predict the probable locations where a user is highly likely to be at the give time.

B. Weighting Vulnerable Time/Location Pairs

We guess the vulnerable locations that the adversary may know and use them as dummy locations. Most social networking sites ask for similar location-related information, specifically current home and office addresses, school, hometown, and more. Therefore, users can predict their vulnerable information based on their disclosure patterns on those websites. Further, the user knows where he/she often goes at a particular time. Our method generates dummy locations for those vulnerable place and time pairs to confuse the attacker. For each vulnerable location and time pair $P(location \cap time)$, our approach multiplies a weight, $risk$, where the $risk > 1$. This means that we assign more emphasis to the location and time pairs that are predicted to be vulnerable, *e.g.*, the times and the locations that the user frequents. As a result, we have (2).

Fig. 3. KAIST campus map.

$$P(dummy) = \frac{P(location \cap time) \cdot risk}{P(time)} \tag{2}$$

- $risk > 1$, if the location/time pair is vulnerable
- $risk = 1$, if the location/time pair is not vulnerable

Finally, we generate dummy locations from the top $P(dummy)$ in (2).

V. Experimental Design and Evaluation

To evaluate our approach, we performe experiments based on real logged data of a person. The experiments are designed to answer the following questions:

Q1 How robust are our dummy locations generated by conditional probabilities against the attack scenario?
Q2 How does weighting vulnerable location and time pairs help ensure more robust dummy locations?

A. Settings

Dataset We construct a dataset that contains logged data of time and location from one target user. The logs were recorded for 17 days, resulting in 263 log data instances in total. We reduce the scale of the locations to places on the KAIST (Korea Advanced Institute of Science and Technology) campus to obtain valid evaluation data within a limited time. We consider 80 popular places that are indicated in Fig. 3. Among the 80 possible places, the logged data contained 10 locations

TABLE I
TEN LOCATIONS ON THE CAMPUS LOGGED BY ONE PERSON

Location	Number of Instances
Building N1	116
Cafeteria (North)	15
Twosome Place cafe	8
Sejong dormitory	109
Cafeteria (East)	1
Fitness center (Sejong)	2
Library	1
Duck pond	1
Cafeteria (West)	4
Coffee Bean cafe	6
Total	263

as specified in Table I. We train with 218 instances and test our method with 45 instances that consists of the last 4 days of data.

Attacker's Scenario We assume that the adversary knows the following information prior to the attack:
User's information (L): We simulate the adversary's attack and search for the user's information on the Internet. As a result, the addresses of the user's office, university, high school, and hometown were retrieved. Among the locations we found, the office (Building N1) and current residence (Sejong dormitory) were locations on the KAIST campus.
Spatiotemporal information (TL): We assume that the adversary knows all the cafeterias and cafes, residences, and office buildings on the KAIST campus. Further, the adversary has general knowledge that people normally go to cafeterias or cafes during mealtimes (7 a.m. – 9 a.m., 11 a.m. – 1 p.m., and 5 p.m. – 7 p.m.), residences to sleep during late night (2 a.m. – 6 a.m.), and works during the afternoon (2 p.m. – 4 p.m.).

The attacker guesses one real location conforming to the strategy (Algorithm 1) mentioned in Section III-C.

Performance Measure We compute the average probability that the attacker correctly identifies the true location(r). According to our attack scenario in Section III-C, the adversary randomly chooses from the possible real location candidates(R) that are selected using prior information about the user and the spatiotemporal knowledge. The success probability of attacker is calculated as in (3). The lower the probability, the more difficult it is for the adversary to identify the true location from the dummy locations, thus, the more resistant the method is to attack.

$$\text{Success Probability of Attacker} = \begin{cases} \frac{1}{|R|} & \text{if } r \in R \\ 0 & \text{if } r \notin R \end{cases} \quad (3)$$

B. Experimental Results

We evaluate robustness against the suggested attack strategy by comparing our method with two other schemes. The baseline approach is the method used in [22] which generates dummy locations randomly, each location having equal probability. The DLS algorithm is introduced in [28], which uses an entropy-based scheme. Additionally, we indicate the theoretical result from the optimal k-anonymity algorithm, in which the probability of guessing the real location is $1/k$.

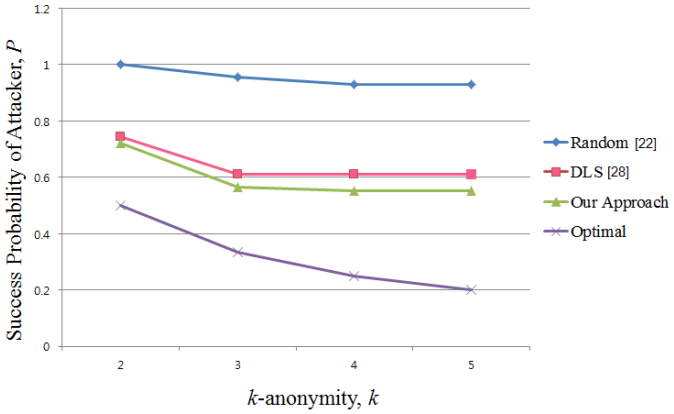

Fig. 4. Comparison of success probability of attacker and different k-anonymity.

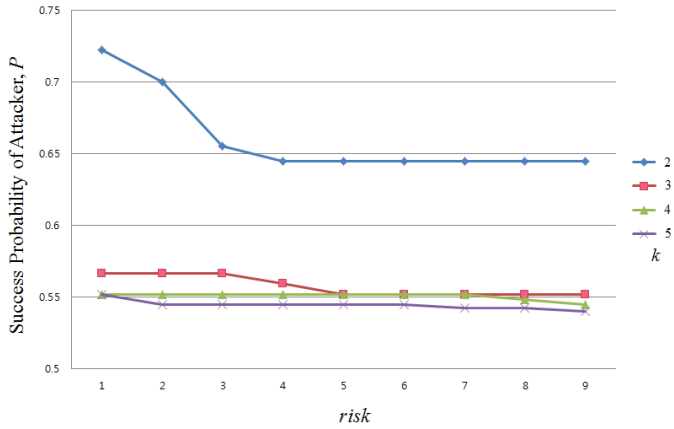

Fig. 5. The performance of our approach according to different $risk$ and k.

Fig. 4 compares the result of our dummy generation method ($risk$=1) and the other two methods. We show results using different k-anonymity by changing the number of dummy locations. The graphs clearly demonstrate that our approach results in lower probability that the attacker correctly identifies the true location, which means our approach is safer in the face of the attacker's suggested strategy. On average, when our method is used, the attacker succeeds 4.6% less often than when the DLS algorithm is used and 35.6% less often than when the random scheme is used. Further, as k (the number of dummy locations) increases, the performance also increases, which is a reasonable result.

Next, we evaluate our weighting scheme for vulnerable time and location pairs which is described in (2). We set vulnerable time/location pairs according to the user's schedule: (2 a.m. – 6 a.m., Sejong dormitory), (8 a.m., Cafeteria (East)), (2 p.m. – 6 p.m. and 8 p.m. – 9 p.m., Building N1), (11 a.m. – 12 p.m. and 17 p.m. – 18 p.m., cafeterias and cafes on the campus)

TABLE II
COMPARISON OF DUMMY LOCATIONS WHEN A QUERY IS GIVEN AT LUNCHTIME.

Real query	Cafeteria (East), 12 p.m.	
Approach	**Dummy 1**	**Dummy 2**
Random [22]	Building E12	Building W5
DLS [28]	Library	Duck pond
Our approach ($risk$=1)	Building N1	Cafeteria(North)
Our approach ($risk$=5)	Cafeteria(North)	Twosome Place cafe

TABLE III
COMPARISON OF DUMMY LOCATIONS WHEN A LOCATION WITH LOW QUERY PROBABILITY IS GIVEN.

Real query	Eoeun Hill, 3 p.m.	
Approach	**Dummy 1**	**Dummy 2**
Random [22]	Fitness center (Areum)	Cafeteria (North)
DLS [28]	Duck plaza	Building E16
Our approach ($risk$=1)	Building N1	Sejong dormitory

The $risk$ weight for vulnerable locations is set from 1 to 9 to test our method. Table 5 demonstrates that our approach using $risk$ weight results in lower probability of attacker's success than the approach using no weight ($risk$=1), meaning that in the former it is more difficult to identify the true location. Moreover, the effect of using $risk$ is clearer when lower k is used: the case where k is 2 shows the greatest improvement. The reason for this is that a large k, meaning many dummy locations, already includes most of the realistic locations we have in the dataset. Thus, we can prove that $k = 3$ is enough for the user in our experiment. On the other hand, we can assert that our approach can be effectively applied to location-based services which lower network cost is preferred (lower k). If an application allows only one dummy location to be sent to the service provider, our approach will generate much safer dummies.

C. Qualitative Analysis

Our method takes time into consideration, which gives more realistic dummy locations. For example, Table II shows two dummy locations generated by different schemes when given a query at 12 p.m. (around lunch time). The random approach [22] generates irrelevant locations inconsistent with both the user's query history and the time. The DLS [28] algorithm generates dummy locations based on query probabilities which do not take the time into account. Since the query, cafeteria (East), has similar query probability to the library and the duck pond, the DLS algorithm generates those locations as dummy locations. Additionally, we observe that this approach often generates cafeteria locations during late-night hours, which is intuitively unreasonable since cafeterias normally do not open at 4 a.m. The reason for this is that cafeterias have high query probabilities and the DLS approach selected them as dummy locations regardless of the time. On the other hand, our approach generates Building N1 (the user's office) and the cafeteria (North) as dummy locations. This means that Building N1 and the cafeteria (North) were the most frequently visited places at 12 p.m. in previous days. The cafeteria (North) is highly likely to be visited around lunch time and Building N1 is where the office of the user is situated. Thus, our dummy locations seem to be more realistic compared to other locations. If $risk$ is set to 5, contextual information is more emphasized and cafeterias and cafe locations are more preferred around lunch time. This results in the cafeteria(North) and Twosome Place cafe, which are also very realistic to an adversary with inference ability.

Furthermore, our approach handles locations with low query probabilities differently. For instance, what kind of dummy locations are generated when the user visits a new place? This situation is simulated in Table III. Eoeun Hill is a new place which the user never has visited before: thus, its query probability is zero. The random approach selects randomly from all possible locations in campus without considering query probabilities. Consequently, the dummy locations that are generated can be irrelevant, such as the fitness center (Areum), or relevant with good luck, such as the cafeteria (North), which is a place that the user visited before. However, the lucky cases are very rare because there are so many places in the area, whereas people normally go to a limited set of places. The DLS approach generates dummy locations that have similar query probability to that of the real location. Therefore, in this case, the method generates places that the user never has visited before, such as the duck plaza and Building E16. All dummy locations and real locations are new to the attacker. However, if there are some locations that are familiar to adversary, the adversary may select familiar ones rather than new locations. This idea is implemented in our approach, which generates places with high query probabilities regardless of whether the real query has low probability. In Table III, our approach generates Building N1 (the user's office) and Sejong dormitory (the user's residence) as dummy locations which are much more likely than the new place, Eoeun Hill. Therefore, our approach can generate more tempting dummy locations compared to other schemes.

In spite of the good aspects mentioned, our approach requires some memory space to save the vulnerable location/time pairs when $risk$ is used. In addition, we focus ourselves on the single-point attack in this research and we show that

our method is simple and effective in such an attack scenario. However, our approach may become vulnerable when a multiple-position attack is executed, which may find patterns from multiple points. We plan to deal with multiple-position attacks in future studies.

VI. Conclusion and Future Work

In this paper, we describe an attacker's strategy in a location-based system where there is an honest-but-curious service provider that has prior knowledge of the user and external background information. To tackle the threat, we introduce a dummy generation method that generates effective dummy locations using conditional probabilities given a particular time. Further, we show that the location privacy is enhanced when considering spatiotemporal information. Experimental results show that our simple statistical method provides more effective dummy locations than other methods.

We demonstrate that our dummy generation method is successful in single-position attack scenario. We plan to extend our method to handle multiple-position attacks. We believe our simple and effective approach can be easily deployed in real-world location-based services to enhance the location privacy.

Acknowledgment

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. 2010-0028631).

References

[1] Foursquare website. [Online]. Available: https://foursquare.com/
[2] Shop alert website. [Online]. Available: http://www.shop-alert.com/
[3] Onstar website. [Online]. Available: https://www.onstar.com/
[4] Accuweather website. [Online]. Available: http://www.accuweather.com/
[5] P. Keikhosrokiani, N. Mustaffa, N. Zakaria, and M. I. Sarwar, "A proposal to design a location-based mobile cardiac emergency system (lmces)," *Stud Health Technol Inform*, vol. 182, pp. 83–92, 2012.
[6] Shopkick website. [Online]. Available: https://www.shopkick.com/
[7] Yelp website. [Online]. Available: http://www.yelp.com/
[8] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *Pervas. Comput.*, vol. 2, no. 1, pp. 46–55, 2003.
[9] L. Liao, D. J. Patterson, D. Fox, and H. Kautz, "Learning and inferring transportation routines," *J. Artif. Intell.*, vol. 171, no. 5, pp. 311–331, 2007.
[10] Y. De Mulder, G. Danezis, L. Batina, and B. Preneel, "Identification via location-profiling in gsm networks," in *Proc. of ACM International Workshop on Privacy in the Electronic Society (WPES)*, 2008, pp. 23–32.
[11] P. Golle and K. Partridge, "On the anonymity of home/work location pairs," in *Pervasive Computing*. Springer, 2009, pp. 390–397.
[12] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady, "Enhancing security and privacy in traffic-monitoring systems," *Pervas. Comput.*, vol. 5, no. 4, pp. 38–46, 2006.
[13] J. Krumm, "Inference attacks on location tracks," in *Pervasive Computing*. Springer, 2007, pp. 127–143.
[14] Y. Matsuo, N. Okazaki, K. Izumi, Y. Nakamura, T. Nishimura, K. Hasida, and H. Nakashima, "Inferring long-term user properties based on users' location history." in *Proc. of ACM International Joint Conference on Artificial Intelligence (IJCAI)*, 2007, pp. 2159–2165.
[15] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: Architecture and algorithms," *IEEE Trans. Mob. Comput.*, vol. 7, no. 1, pp. 1–18, 2008.
[16] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proc. of ACM International Conference on. Mobile Systems, Applications, and Services (MobiSys)*, 2003, pp. 31–42.
[17] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady, "Preserving privacy in gps traces via uncertainty-aware path cloaking," in *Proc. of ACM Conference on Computer and Communications Security (CCS)*, 2007, pp. 161–171.
[18] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, "Preventing location-based identity inference in anonymous spatial queries," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 12, pp. 1719–1733, 2007.
[19] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, "The new casper: query processing for location services without compromising privacy," in *Proc. of ACM International Conference on Very Large Data Bases (VLDB)*, 2006, pp. 763–774.
[20] J. Freudiger, R. Shokri, and J.-P. Hubaux, "On the optimal placement of mix zones," in *Privacy enhancing technologies*. Springer, 2009, pp. 216–234.
[21] B. Hoh and M. Gruteser, "Protecting location privacy through path confusion," in *Proc. of IEEE International Conference on Security and Privacy in Communication Networks (SECURECOMM)*, 2005, pp. 194–205.
[22] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in *Proc. of IEEE Internation Conference on Pervasive Services (ICPS)*, 2005, pp. 88–97.
[23] M. Wernke, P. Skvortsov, F. Dürr, and K. Rothermel, "A classification of location privacy attacks and approaches," *Pers. Ubiquit. Comput.*, vol. 18, no. 1, pp. 163–175, 2014.
[24] J. Krumm, "Realistic driving trips for location privacy," in *Pervasive Computing*. Springer, 2009, pp. 25–41.
[25] R. Chow and P. Golle, "Faking contextual data for fun, profit, and privacy," in *Proc. of ACM International Workshop on Privacy in the Electronic Society (WPES)*, 2009, pp. 105–108.
[26] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *Proc. of IEEE Symposium on Security and Privacy (SP)*, 2011, pp. 247–262.
[27] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, and J.-Y. Le Boudec, "Protecting location privacy: optimal strategy against localization attacks," in *Proc. of ACM Conference on Computer and Communications Security (CCS)*, 2012, pp. 617–627.
[28] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2014, pp. 754–762.
[29] L. Sweeney, "k-anonymity: A model for protecting privacy," *Int. J. Uncertain. Fuzz.*, vol. 10, no. 05, pp. 557–570, 2002.