TIS-BAD: A TIME SERIES-BASED DEOBFUSCATION ALGORITHM

TIS-BAD: UN ALGORITMO DE SERIE DE TIEMPO BASADO EN DESOFUSCATION

Recibido: 20 de septiembre de 2014 - aceptado: 10 de diciembre 2014

Miguel Labrador.[1]
University of South Florida

Pedro. Wightman[2]
Universidad del Norte

A. Santander[3]
Universidad del Norte

Daladier. Jabba[4]
Universidad del Norte

Miguel. Jimeno[5]
Universidad Del Norte

Keywords:
Location obfuscation; Privacy; Deobfuscation; LBIS

Abstract

This paper presents a formal definition of a deobfuscation technique for noise-based obfuscation algorithms called TIS-BAD (Time Series - Based Deobfuscation) which implements an exponentially weighted moving average over the obfuscated data to filter the induced noise. In the literature there have been very few efforts to present such formal deobfuscation techniques, being this is one of the main contributions of this work. We evaluate the TIS-BAD algorithm against the Rand and N-Rand obfuscation algorithms, including both location and time scrambling, for straight and non-straight routes. The results show that the TIS-BAD algorithm can filter from 47% to 60% of the induced noise by the obfuscation algorithms, reducing considerably the protection on the users' location information.

Palabras clave:
Ofuscación de ubicación, privacidad, LIBS

Resumen

En artículo presenta una definición formal de una técnica desofuscación de algoritmos de ofuscación relacionadas con el ruido llamados TIS-BAD (Time Series - Based Deobfuscation) que implementa un promedio móvil ponderado exponencialmente en los datos ofuscado para filtrar el ruido inducido. En la literatura se han realizado muy pocos esfuerzos para presentar este tipo de técnicas formales de desofuscación, siendo esta es una de las principales contribuciones de este trabajo. Evaluamos el algoritmo TIS-BAD en contra de los algoritmos de ofuscación Rand y N-Rand, incluyendo tanto la ubicación como el tiempo de aleatorización, para las rutas rectas y no rectas. Los resultados muestran que el algoritmo de TIS-BAD puede filtrar de 47% a 60% del ruido inducido por los algoritmos de ofuscación, reduciendo considerablemente la protección de la información de ubicación de los usuarios.

1. Department of Computer Science and Engineering, University of South Florida, Tampa, FL, USA. E-mail: mlabrador@usf.edu
2. Departamento de Ingeniería de Sistemas. Universidad del Norte. Barranquilla, Colombia. E-mail: pwightman@uninorte.edu.co
3. Departamento de Ingeniería de Sistemas. Universidad del Norte. Barranquilla, Colombia. E-mail: asantand@uninorte.edu.co
4. Departamento de Ingeniería de Sistemas. Universidad del Norte. Barranquilla, Colombia. E-mail: djabba@uninorte.edu.co
5. Departamento de Ingeniería de Sistemas. Universidad del Norte. Barranquilla, Colombia. E-mail: majimeno@uninorte.edu.co

*Este artículo es asociado al proyecto de investigación: TIS-BAD: A Time Series-Based Deobfuscation Algorithm

I. INTRODUCTION

A Location-Based Service can be defined as an application that provides users with information based on their geographical position, which could be obtained from the mobile device they are carrying and from which they are accessing the service, or using a manually defined location [1].

Location-Based Services (LBSs) and Location-Based Information Systems (LBISs) have shown a growth trend in the last years due to factors like the development of faster cellular- based Internet communication technologies, the popularization of GPS-based navigation both offered via web through smart- phones or in devices installed in the vehicles, and the availability of open source APIs to work with maps, geographic databases, and GIS applications. This list of advances has inspired the development of a great number of LBSs all around the world. Comments like the next one from the European GNSS Agency confirm the potential of this technology: "LBSs are at the start of an impressive growth curve, possibly resulting in countless opportunities for entrepreneurs and a €100bn market" [2].

This opportunity also brings some associated risks. Service providers now have the location information of their clients, which could be used for other purposes than the one initially informed to the user. For example, users can be tracked and their preferences known by service providers who may send spam and undesired advertisement back to users. Even worse, the information could be stolen during a security breach and used later in actions that may put the users in danger. One well- known technique to protect the location information of the users is Location Obfuscation (LO).

Location obfuscation consists of applying a non-reversible alteration to the exact location of the users, so that it does not reflect their exact location but still contains enough information about their geographical position to allow the provider to answer users' queries. The most common LO techniques are the noise-based techniques. These techniques add random noise to the location information to generate the altered one, most of the times using symmetrical random distributions like uniform and Gaussian, or asymmetric, like exponential. However, in the literature, there has not been enough work where deobfuscation techniques are formally defined and evaluated against these obfuscation algorithms.

This works presents the TIS-BAD deobfuscation algorithm, a time series-based technique that uses exponentially weighted moving average filter to reduce the induced noise in an obfuscated path in order to reveal morphological information on the user's route that can lead to a full identification of the protected path. TIS-BAD is evaluated along two noise-based obfuscation methods, Rand and N-Rand using two different kinds of paths. In addition, these two algorithms are extended in this paper by adding time obfuscation as a possible tool against the TIS-BAD algorithm.

The results show that the TIS-BAD can filter out between 47% and 60% of the induced noise on the obfuscated paths, depending on the type of path and the obfuscation technique. Based on the experiments with different parameters for the TIS-BAD algorithm, the best configurations appear to need a small number of historic values, a high impact on recent obfuscated values and a balanced proportion between the averages of obfuscated and estimated data. Most configurations with these qualities showed the lower levels of distance between the estimated points and the original ones.

The paper is organized as follows: Section 2 presents a summary of related work. Section 3 describes the time-series deobfuscation technique. Section 4 shows the design of the experiments and the results of the performance evaluation. Finally, Section 5 presents the conclusions and future work.

II. RELATED WORK

Privacy and security for the user's location have been an active topic of research for the last few years, as a way to protect the location information of the users, which may be at risk from attackers who would like to access exact information about the mobility patterns of the users.

There are different ways to provide location privacy. Some of the most important are anonymity, query-based and location obfuscation techniques.

Anonymity is a technique that separated the location information from its owner, in order to prevent the traceability of the location information; this is, the users report their location information and it is stored without any relation with the user that generated it. The location information associated to the user is chosen among other preexisting data in the database, so that the reported location is not the actual one. Some techniques use pseudonyms and the use of third party location providers [3], and with k-anonymity techniques in which a user's exact location cannot be distinguished among k-1 other user [4], [5]. Another technique reveals only non-critical information or incomplete information about the routes such that there are not enough details about the route taken by the user for it to be reconstructed [6].

The query-based techniques approach privacy by protecting the process in which the user access the LBIS, and avoid revealing the location of the user. Most of these techniques do not store the location of the user, thus they would not be useful for tracking applications, so they are not in the scope of this paper. Some of the techniques in this category are [7].

Location obfuscation can be defined as "the means of deliberately degrading the quality of information about an individual's location in order to protect that individual's location privacy." [8] The main idea of is for the original location to be altered or generalized in order to hide the exact location of user, while the LBS service provider still can provide a desirable level of service. Some of these techniques can be parameterized by the user in order to find a level of compromise of its location information in order to obtain a better service. Many different location obfuscation techniques have been proposed. In [9, 10] the authors propose adding random noise to existing data in order to hide their actual content.

In [11], the authors generalize the location of the user, redefining possible areas of location, in which no location point is provided but an area that contains the actual location. In [12], the authors propose rounding the location based on a predefined set of landmarks or grid cells. In [13] the authors propose the generation of alternate dummy paths so the attacker could not recognize the actual path of the user. Other schemes use combined solutions including obfuscation and anonymity [4].

However, in the literature there have not been great efforts in the formal definition of deobfuscation techniques. This is a very critical area in privacy research in order to test the performance of the obfuscation techniques against possible tools that attackers could use to reverse the obfuscation to obtain the original location information. In addition, if the exact location of not possible to recreate exactly, at least a cleaner view of a user's path that may allow attackers to infer the users' actual route after further analysis based on the real map of the area where the route is from.

Most works that include deobfuscation algorithms always present them jointly with their obfuscation techniques. For example, in [9] the authors propose an attack scheme based on trying to identify the random distribution used to generate the noise in the data so they can classify more accurately the information. In [11] the authors define a more empirical approach in which attackers blindly select different options of area modification and then the authors quantify how good this guessing process could be. In [13, 14] the authors propose Markov-based approaches in which the locations on a user can be discretized in order to obtain the best most probable path of the user.

This paper introduces the TIS-BAD algorithm, a deobfuscation algorithm that uses time series analysis using an exponentially weighted moving average (EWMA) filter in order to eliminate the induced noise in the route, so a much "cleaner" of the obfuscated path can be calculated. This technique will be described with more details in Section 3.

III. TIME SERIES-BASED DEOBFUSCATION TECHNIQUE

In general, most noise-based location obfuscation techniques consider each point as an individual piece of information that gets transformed by adding random noise to it. Each point represents an occurrence of a random value of the chosen random distribution, which, if not configured properly, can ease the process of estimation of the original paths.

For example, the nature of the noise generated using symmetrical random distributions like uniform or Gaussian is usually centered at 0, which becomes its expected value in the long run. This fact allows the use of the moving average technique to estimate the original path from an obfuscated one because, from a set of random occurrences it is expected that the sample mean will add up all the noise, canceling out its effect on the location information, thus reveal the original trace.

This technique may not reveal the actual exact corresponding location to the obfuscated point, but it will filter most of the induced noise and reveal a morphologically equivalent path that should facilitate the process of exactly estimating the location of the user.

The attack model considered in this work is the one in which the attacker is able to gather obfuscated traces of routes from a single user. The information has a single pseudonym per trace, so all the points of a user could be queried all together. No other privacy techniques are considered, like anonymity, cloaking or query-based algorithms. The attacker wants to estimate a more detailed location of the user from the obfuscated data.

A. Deobfuscation formula

The TIS-BAD deobfuscation algorithm uses an exponentially weighted moving average filter over sequence of obfuscated points P' and a sequence of estimated points P^* to estimate a representative point of a set of k locations. Both the averages of the actual obfuscated points and the estimated points are later

added in a linear combination formula to estimate the next point in the series. Eq. 1 shows the definition of the formula, where *k* is the number of previous values from both sequences that are used in the time series formula, α is the weight parameter for the obfuscated points, δ is the weight parameter for the previously estimated points, and β is the linear combination parameter for both average points estimated from the sequences.

$$P_i^* = \beta \cdot \frac{\sum_{j=1}^{k} P'_{i-j} \cdot \alpha^j}{\sum_{j=1}^{k} \alpha^j} + (1-\beta) \cdot \frac{\sum_{j=1}^{k} P^*_{i-j} \cdot \delta^j}{\sum_{j=1}^{k} \delta^j}. \quad (1)$$

The algorithm uses the first k obfuscated values raw, and assumes those same values for the estimated ones. From the *k*+1 value on, the TIS-BAD algorithm starts applying the formula in order to estimate the location point.

The parameters of the formula have a great impact on the results of the estimation. The k parameter defines how many values from the past will be considered. In a very straight path, it is expected that the average of more values will generate an estimate closer to the original path, while in a very curvy path including many points may misguide the algorithm into areas where the path does not go by exactly.

On the other hand, both α and δ define the rate of impact of the historical values in the estimation, of the obfuscated values and the estimated ones. A large value in these parameters means a more homogeneous distribution of the impact among all *k* values, while a small value assigns a large impact to the first value and the impact decreases rapidly for the rest of the *k* past values.

The β parameter defines the balance between the average of the obfuscated values and the average of the estimated values. The average of the obfuscated values guides the algorithm to follow the general trend of the path, while the average of the estimated values includes an element of noise reduction and filters the variability from the obfuscated values; in other words, it balances the new information obtained from the obfuscated values with the trend previously found in the data.

Fig. 1 shows an example of the effectiveness of the TIS-BAD algorithm. The scenario has 500 points with an *r* parameter of 200 meters and obfuscated by the N-Rand algorithm with N=10 and a curvy design. The original path is shown in black in the middle of the figure. It can be identified thanks to its straight sections. The route with a lighter color and that occupies all the back of the figure is the obfuscated path, which illustrates the large amount of noise induced in this scenario and the impossibility to visually identify the original path. The blue (solid darker gray) line represents the estimation calculated by the TIS-BAD algorithm. It can be seen that the estimated path does not corresponds directly to the original path, but it considerably reduced the noise and produced a morphologically close structure that could be used to infer more precisely the whereabouts of the user. The deobfuscated path used in this experiment uses the parameters *k*=10, α=0.7, δ=0.5 and β =0.3.

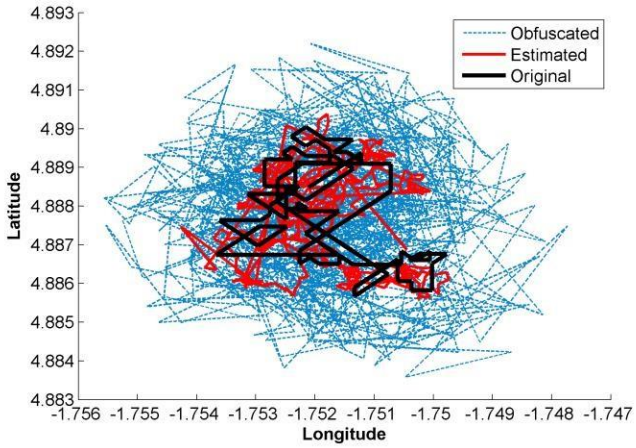


Fig. 1. Example of the application of the TIS-BAD algorithm over an obfuscated path.

The next section is dedicated to the evaluation of the TIS- BAD algorithm in terms of noise reduction from the obfuscated paths, measured as the average distance from the original points to the new estimated points. A large set of parameter combinations is considered in the experiments in order to identify the best configurations.

IV. PERFORMANCE EVALUATION

This section presents the evaluation of the TIS-BAD algorithm applied over several scenarios in order to determine its general performance. The experiment includes the following factors on its study: type of mobility pattern, obfuscation algorithm, and internal parameters of the TIS-BAD algorithm.

The measured performance variable is the average distance from the original location to the deobfuscated one, and its standard deviation. This will be contrasted also with the average distance provided by the obfuscation algorithm. The results are obtained from the average of 50 paths following the same mobility pattern. These paths are obfuscated using the algorithms and their time-obfuscation variants. Then, each obfuscated path is deobfuscated using

each configuration, and a grand average per configuration is calculated in order to obtain the average distance provided by that configuration of the TIS-BAD algorithm.

A. *Mobility patern*

This factor was included in the study in order to determine the level of impact of the mobility pattern of the user on the way the algorithm works. It would be expected that a very curvy path will generate more problems to the algorithm than a straighter route.

Based on this idea, two types of mobility patterns were included: mostly straight and mostly curvy. The paths were synthetically generated using a Markov-based automaton.

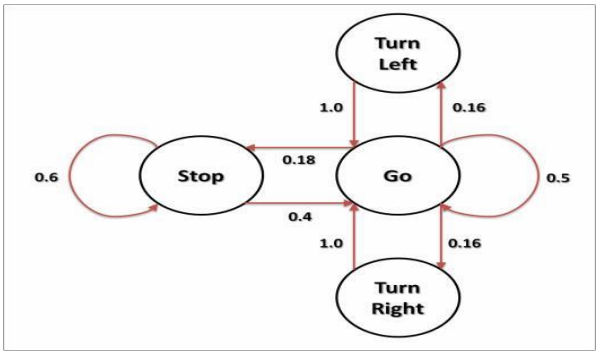


Fig. 2. Example of the Markovian automaton to generate paths for the semi-straight mobility pattern.

The automaton, as shown in Fig. 2 considers 4 states: Go, which means that the trajectory will continue based on the same direction defined in the previous state, Turn Right and Turn Left, in which the orientation will change and the trajectory will advance in the new orientation, and Stop which considers the user holding its current position for a certain amount of time. The changes in distance on each step is equivalent to moving 10 meters in the selected orientation, but the time difference between steps is randomly generated between 0.7 and 1.7 seconds, which should model changes in speed in the user's movement in a car at between 50 and 20 kilometers per hour. The changes in direction were discretized based on 45 degree steps, which means a turn could be 0, 45, 90, 135, 180 degrees on the respective direction of the state, plus a displacement in that new direction following the same description of the Go state. In the case of the Stop state, the waiting time in that state is randomly generated between 0 and 30 seconds. From the state machine it can be seen that a user can have more than one waiting period before it can go to the Go state again. The probability of staying in the Stop stage is 60% in all the scenarios.

The transition probability matrix is used to define the probability of staying in one stage or changing to another one. Table 1 shows a summary of the transition probabilities used in the generation of both semi-straight and curvy routes. All routes were assumed to start at the coordinate (Lat=4.887933, Long=-1.752478), which us a certain real location from Takoradi, Ghana.

B. *Obfuscation factor*

Four factors were defined on the obfuscation process: Algorithm, radius, the N parameter and the application of time obfuscation. The Rand and N-Rand obfuscation algorithms were selected based on their performance evaluation in [16]: the first one produces a large average distance between the original path and the obfuscated one, and the second has the lowest complexity of all obfuscation techniques.

Each of the paths generated for this experiment were obfuscated by these algorithms and the combination of their parameters. The chosen parameters for the experiments were a noise radius *r* of 100 meters: 100 meters is the size of a typical block of Hispanic colonial cities. In addition, time obfuscation is also evaluated based on a single value for the rt parameter. This parameter is kept with a single value for this work but may be studied with more detail in a future work. The N parameter determines the number of random samples in order to select the largest one in the N-Rand algorithm; in other words, a larger N will generate larger average distances. A value of N=10 generates the largest distance between the original and the obfuscated location, but it does not produce a high variability, while N=4 produced a shorter distance but a higher variability. It is expected that variability may have an impact on the ability of the deobfuscation algorithm to estimate the real location.

C. *Deobfuscation factor*

Each of the parameters on the TIS-BAD deobfuscation technique has 5 different levels, and each of the obfuscated files is processed in order to estimate the original path using the 125 combination of parameters. As mentioned before, α and δ does have an impact on the weight of the historical values in both obfuscated and estimated values, and β plays as the balancing element between these estimations.

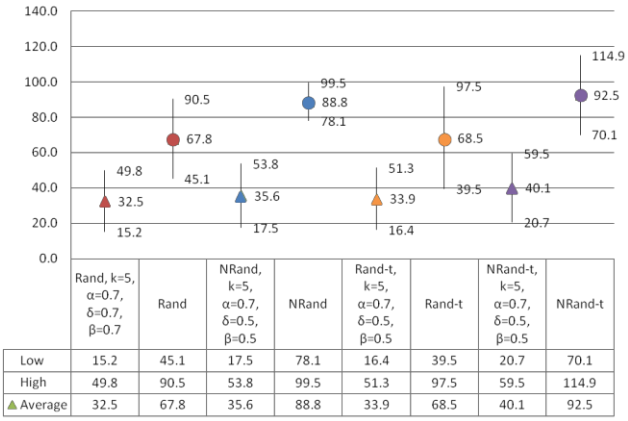
Due to the large number of experiments, the results will show only the configurations that produced the estimation with the lowest average distance between the points in the original route and the points in the estimated path. The different factors and levels of the experiments are summarized in Table I.

TABLE I. TABLE TYPE STYLES

Factor	Level	Description
Mobility Patterns	Semi-Straight	Go→Go=0.50, Go→Left=Go→Right=0.16, Go→Stop=0.18
	Curvy	Go→Go=0.90, Go→Left=Go→Right=0.03, Go→Stop=0.04
Obfuscation algorithm	Rand	r = 100 r_t = 10 secs W/ and W/O time obfuscation
	N-Rand	N = 4, 10 r = 100 r_t = 10 secs W/ and W/O time obfuscation
TIS-BAD parameters	k	5, 10, 15
	α	0.1, 0.3, 0.5, 0.7, 0.9
	δ	0.1, 0.3, 0.5, 0.7, 0.9
	B	0.1, 0.3, 0.5, 0.7, 0.9

D. Results

The results are divided into 2 figures, defined by the type of mobility pattern of the path. Each figure contains the results of the configuration that produced the lowest average distance of the estimation from the original path, including also their standard deviation, for 4 scenarios defined by the obfuscation algorithm (Rand and N-Rand) and the application of time obfuscation. They also include the average distance produced by the obfuscation algorithms in order to illustrate the improvement in distance provided by the TIS-BAD deobfuscation algorithm.



	Rand, k=5, α=0.7, δ=0.7, β=0.7	Rand	NRand, k=5, α=0.7, δ=0.5, β=0.5	NRand	Rand-t, k=5, α=0.7, δ=0.5, β=0.5	Rand-t	NRand-t, k=5, α=0.7, δ=0.5, β=0.5	NRand-t
Low	15.2	45.1	17.5	78.1	16.4	39.5	20.7	70.1
High	49.8	90.5	53.8	99.5	51.3	97.5	59.5	114.9
▲Average	32.5	67.8	35.6	88.8	33.9	68.5	40.1	92.5

Fig. 3. Average distance in curvy paths with N=4 and R=100

The first scenario is the curvy scenario where the fast changes in direction are expected to become a problem for the TIS-BAD algorithm. Fig. 3 shows the results for these scenarios.

From these results, some conclusions can be drawn. The first and more important one is that the TIS-BAD evidently reduces the distance between the obfuscation paths and the original paths. This reduction is about 52% over the Rand protocol and about 60% over the N-Rand protocol without time obfuscation, and about 50% and 57% for Rand and N-Rand with time obfuscation.

Also, it can be seen that time obfuscation does not have a great impact on the average noise to the obfuscated paths compared to the version without it, but it increases the variability of the noise, producing a slight improvement of the obfuscation algorithms against the TIS-BAD algorithms, however, based on the T test for paired mean comparison, they are proves statistically different with 99% confidence in both scenarios. This means that the N-Rand indeed produces better absolute deobfuscation than the Rand algorithm because it keeps a longer distance from the original points and the estimated ones.

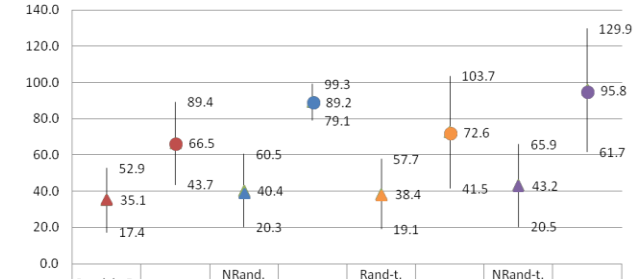
One interesting results is that even though the N-Rand produces a larger distance from the original path, it is more sensitive to the time series-based attack, allowing a higher filtering ratio; this is, the N-Rand generates considerably larger noise than the Rand algorithm, but the estimated path is very similar to the one obtained from Rand. For some applications, a 7% filtering ratio difference may imply that the Rand algorithm may be a better option for obfuscation due to its lower complexity and better resistance against deobfuscation; however, the average distance, as mentioned before, is always higher for the N-Rand algorithm thus this algorithms provides a better absolute protection. The sensitivity of N-Rand may be related to the variability of the noise generated by the obfuscation algorithms, i.e., the standard deviation of N-Rand is 0.75 and 0.5 of the one from Rand, with and without time obfuscation, respectively. This relation between distance and variability can be seen in the 4 meters difference in average distance of the N-Rand algorithm when time obfuscation was included. The best configurations of the TIS-BAD algorithm have some things in common for all the scenarios: α parameter is 0.7 in all configurations, and δ and β are between 0.5 and 0.7, with the similar values in the scenarios.

A large α parameter means that the weight of the historical values does not decay very fast, but their impact indeed starts decreasing as k increases. In the case of δ, it tends to have a value of 0.5 which gives lower weights to the historical values, giving a higher weight on the newer values. The β parameter tends to give equal priority to both averages from the obfuscated paths and from the estimated values. This confirms that fact that the average of the estimated value helps filtering the noise from the obfuscated one. In addition, the lowest distances were obtained with k=5, which is the configuration that uses the least

number of historical values. This was not completely expected from a semi-straight path, in which more nodes may help filtering the noise better, but on the other hand, a larger set of historical points may increase the average distance due to the fact that the average point will include many more points, some of them very distant to the original one that is being estimated, thus the average point will be also farther from the original one.

The second scenario, the semi-straight paths, uses a less changing path course in which the user stays longer in the same direction. This is expected to have a lower impact on the performance of the TIS-BAD algorithm because the average of the values may include sets of points in similar directions. Fig. 4 shows the results of the average distance obtained with the deobfuscation algorithm.

The results show similar general behavior with the previous scenario. The N-Rand algorithm presents a better obfuscation solution in terms of distance between the original points and the estimated ones, thus protecting the original path. Contrary to the expectation, the nature of the straighter paths generated slightly larger average distance in all scenarios compared to the semi-straight paths. This can be appreciated also in the reduction of the filtering ratio to values of 47% and 54% of the noise for the Rand and N-Rand algorithms, respectively. In other words, the TIS-BAD algorithm could not filter as much noise as it did on the curvy paths. This behavior in the straighter paths can be explained by the fact that, even though the curvy paths have many more changes in directions, points tend to be closer together in the curvy paths, thus the average distance among points in a curve is less than the average distance over a straight line. The difference between the point dispersion between paths can be seen clearly in Fig. 4.



	Rand, k=5, α=0.7, δ=0.1, β=0.7	Rand	NRand, k=5, α=0.7, δ=0.9, β=0.7	NRand	Rand-t, k=5, α=0.7, δ=0.3, β=0.7	Rand-t	NRand-t, k=5, α=0.7, δ=0.3, β=0.5	NRand-t
Low	17.4	43.7	20.3	79.1	19.1	41.5	20.5	61.7
High	52.9	89.4	60.5	99.3	57.7	103.7	65.9	129.9
▲Average	35.1	66.5	40.4	89.2	38.4	72.6	43.2	95.8

Fig. 4. Average distance in semi-straight paths with N=4 and R=100

Another interesting result is that in this scenario the filtering ratio was almost exactly the same with and without time obfuscation, thus the time obfuscation did not have any impact in protecting the location information better, despite the evidently increase on the variability of the noise compared with the semi-straight paths. This behavior can be explained by the fact that the noise was so much higher than the distance between the nodes that the noise radius may absorb to some level the impact of the time scrambling.

V. CONCLUSIONS AND FUTURE WORK

Privacy in location-based information systems is a very important area of research, looking toward a broad adoption of these kinds of services in everyday tasks like navigation, shopping, security, telemedicine, etc.

Noise-based location obfuscation algorithms are one of the most common ways to provide a certain level of protection through the addition of random noise to the real location being reported to the LBS provider. On the other hand, very small efforts have been done in the area of deobfuscation in order to formalize techniques that could help testing new algorithms.

This work presents the TIS-BAD deobfuscation algorithm, a time series-based algorithms that is able to filter up to 60% in obfuscated paths generated with the Rand and the N-Rand obfuscation algorithms. These results show that some parameters of the TIS-BAD algorithm tend to have a common behavior among most scenarios, like a α parameter with 0.7, and a β parameter with 0.5 and a δ parameter between 0.3 and 0.5, which means that the protection provided by noise-based obfuscation may be vulnerable to the out-of-the-box attacks, or even worse, a brute force analysis of the obfuscated data with a large set of configuration combinations may show the best configuration to use in order to reduce the noise.

In general, the N-Rand algorithm provided better absolute protection, despite the fact that it could be filtered easier than the Rand algorithm. This fact could be explained by the lower variability that the N-Rand algorithm has compared to the Rand algorithm. The results also showed that the worst case scenario for the TIS-BAD algorithm is when the path is semi- straight and time obfuscation is included, which increases the variability of the error. The best results are obtained when the location points are close by because the average function filters a larger portion of the noise. This means that if the user does not move very fast or stays in a similar area, it is possible to narrow down their location more efficiently. Both these elements can be used in order to create a new noise-based algorithm that can reduce the filtering capabilities of the TIS- BAD algorithm.

VI. REFERENCES

[1] A. R Pérez, M. Labrador and P. Wightman. Location-Based Information Systems. CRC Computer and Information Science Series. CRC Press, USA, 2011.

[2] European GNSS Agency. Opportunities Abound in Growing Location- Based Services Market. In http://www.gsa.europa.eu/go/news/opportunities-abound-i

[3] H. Muller T. Rodden, A. Friday and A. Dix. A Lightweight Approach to Managing Privacy in Location-Based Services. Technical Report Equator-02-058. CSTR-07-006, University of Nottingham and Lancaster University and University of Bristol, 2002.

[4] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In Proceedings of MobiSys. LCNC 3468/2005, pages 31–42, 2003.

[5] W.G. Aref, C.Y. Chow, M.F. Mokbel and G. Walid. Casper*: Query processing for location services without compromising privacy. ACM Transactions in Database Systems, 34:1–48, December 2009.

[6] B. Hoh, H. Xiong M. Gruteser, and A. Alrabady. Preserving privacy in gps traces via uncertainty-aware path cloaking. In Proceedings of the 14th ACM conference on Computer and communications security, pages 161–171, New York, NY, USA, 2007. ACM.

[7] C. Bettini D. Riboni, L. Pareschi and S. Jajodia. Preserving anonymity of recurrent location-based queries. In Proceedings of International Symposium on Temporal Representation and Reasoning, pages 62–69, 2009.

[8] L. Kulik M. Duckham and A. Birtley. A Formal Model of Obfuscation and Negotiation for Location Privacy. In Proceedings of Pervasive - LCNC 3468/2005, pages 243–251, 2005.

[9] R. Agrawal y R. Srikant, Privacy-preserving data mining. ACM Sigmod Record, 29:439-450, 2000.

[10] P. Wightman, W. Coronell, D. Jabba, M. Jimeno and M. Labrador. Evaluation of Location Obfuscation Techniques for Privacy in Location Based Information Systems. In Proceedings of IEEE Latincom, pages 1– 6, 2011.

[11] C.A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati. An Obfuscation-Based Approach for Protecting Location Privacy. IEEE Transactions on Dependable and Secure Computing, 8(1):13–27, 2011.

[12] A. J. Brush, J. Krumm, and J. Scott. Exploring end user preferences for location obfuscation, location-based services, and the value of location. In Proceedings of the 12th ACM international conference on Ubiquitous computing, pages 95–104, 2010.

[13] K. Minami and N. Borisov, Protecting location privacy against inference attacks. In Proceedings of the 9th annual ACM workshop on Privacy in the electronic society, pages 123-126, 2010.

[14] R. Shokri, G. Theodorakopoulos, J. Y. Le Boudec, J. P. Hubaux, and E. LCA, Quantifying location privacy. In IEEE Symposium on Security and Privacy (S&P), 2011.

[15] S. Drape and I. Voiculescu. Creating transformations for matrix obfuscation. In Jens Palsberg and Zhendong Su, editors, Static Analysis, volume 5673 of Lecture Notes in Computer Science, pages 273–292. Springer Berlin / Heidelberg, 2009.

[16] S. Drape and I. Voiculescu. The Use of Matrices in Obfuscation. Technical Report CS-RR-08-12, Oxford University Computing Laboratory, 2009.

[17] J. Krumm. Inference Attacks on Location Tracks. In Proceedings of Pervasive, pages 127–143, 2007.