Mix Zones: User Privacy in Location-aware Services

Alastair R. Beresford and Frank Stajano
Laboratory for Communication Engineering, University of Cambridge,
JJ Thomson Avenue, Cambridge. CB3 0FD
[arb33,fms27]@cam.ac.uk

Abstract

Privacy of personal location information is becoming an increasingly important issue. This paper refines a method, called the mix zone, developed to enhance user privacy in location-based services. We improve the mathematical model, examine and minimise computational complexity and develop a method of providing feedback to users.

1. Introduction

Traditionally, privacy of personal location information has not been a critical issue but, with the development of location tracking systems capable of following user movement twenty-four hours a day and seven days a week, location privacy becomes important: records of everything from the shelves you visit in the library to the clinics you visit in a hospital can represent a very intrusive catalogue of data.

Location privacy is an important new issue and several strategies have been suggested to protect personal location information. The first strategy is to restrict access. The Geographic Location/Privacy (Geopriv) Working Group [1] have outlined an architecture to allow users to control delivery and accuracy of location information through rule-based policies. Hengartner and Steenkiste [2] describe a method of using digital certificates combined with rule-based policies to protect location information.

An alternative approach is to degrade information in a controlled way before releasing it. Gruteser and Grunwald reduce the resolution of location information available to location-aware applications [3]. In previous work [4] we introduced the *mix zone* model: the model anonymizes user identity by restricting the positions where users can be located. The model provides:

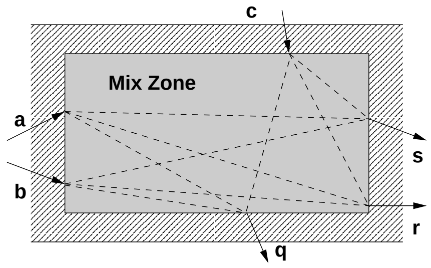
- a middleware mechanism to provide anonymised location information to third-party applications, and
- a quantitative run-time estimate of the level of anonymity provided by the middleware with a particular set of applications.

To gain a proper understanding of the privacy properties of mix zones it is important to find out how hard it is to break the anonymity the system provides. The mix zone approach for calculating anonymity does this: the degree of success in playing the role of attacker—attempting to recover the long-term user identities hidden by the constantly changing pseudonyms—is an inverse measure of the anonymity offered by the system. In this paper we refine and extend our work on the mix zone model to:

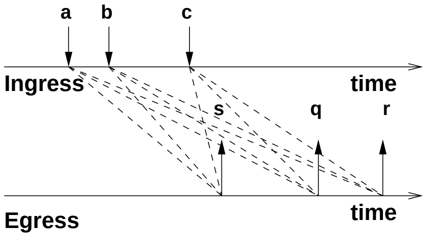
- show how to deal with irregularly-shaped zone boundaries and how to improve the accuracy of the observations for zones of a given size and shape;
- examine and reduce the computational complexity of the algorithm used by the attacker to break the anonymity, and
- develop a method of measuring and providing feedback of the level of anonymity the user experiences.

2. The mix zone model

This section summarises the mix zone model described more fully in [4]. The model assumes the existence of a trusted middleware system, positioned between the underlying location system(s) and untrusted third-party applications. Applications register interest in a geographic space with the middleware; we refer to this space as an *application zone*. Example spaces include hospital grounds, university buildings or a supermarket complex. Users register interest in a particular set of location-aware applications and the middleware limits the location information received by applications to location sightings of registered users located inside



(a) plan view of the mix zone.



(b) Timeline of movements.

Figure 1. Example movement of three people through a simple mix zone. Who went where?

the application zone. Each user has one or more unregistered geographical regions where no application can trace user movements; we call such areas *mix zones*, because once a user enters such a zone, user identity is mixed with all other users in the mix zone, as will become clearer shortly. A *boundary line* is defined as the border between a mix zone and an application zone.[1]

Applications do not receive a traceable user identity associated with a location sighting, but instead receive a pseudonym. The pseudonym allows communication between user and application; such communication must pass through a trusted intermediary to prevent trivial linking of a pseudonym with an underlying user identity. The pseudonym of any given user changes whenever the user enters a mix zone.

The aim of the mix zone model is to prevent tracking of long-term user movements, but still permit the operation of many short-term location-aware applications. We have shown in our previous paper how some more obvious pseudonymity mechanisms can be trivially defeated, allowing identification of the user behind each pseudonym.

Since third-party applications are untrusted they may collude, therefore all third-party application providers are treated as one combined global hostile observer. How well can this attacker correlate movements of users into the mix zone with movements into a subsequent application zone? In other words, can the attacker link together user pseudonyms, and therefore track long-term user movements? Figure 1 provides an example scenario.

An attacker may be able to use historical data from nearby application zones or analytical methods[2] to infer likely user movement across the mix zone. User exit boundary line and time is often strongly correlated to user entry boundary line and time. For example, two users walking down a corridor in opposite directions are much more likely to continue in the same direction than turn around and retrace their steps.

The middleware can use historical data from inside the mix zone to provide a (probably far superior) model of user movement; movement patterns are not time-invariant, but are likely to be self-similar over short time spans, after twenty-four hours and after seven days. A movement matrix is generated to record the frequency of ingress and egress points for different time periods across the mix zone at different times in the day or week. Such a movement matrix provides an upper bound on the accuracy of the model the attacker can generate.

A location system scenario (e.g. E911-enabled cellular phones in an urban setting) can therefore be examined using the mix zone model; the model takes into account the geometry of the zones, the temporal and spatial resolution of the sightings, and the statistical behaviour of the user population to provide a quantitative assessment of how well a hostile observer is able to de-anonymize individual users. When designing a system this analysis can be run iteratively, changing the layout of the zones or the resolution of the sightings until the desired level of anonymity is achieved.

3. Fine-grained modelling of boundaries between zones

The example computational procedure outlined in our previous paper only examines *which* boundary line was crossed and not the specific position of the crossing

[1] Location is a two-dimensional position in this paper, but a more complex model could be developed by moving to three dimensions and considering boundary surfaces rather than lines.

[2] One analytical approach could estimate likely movements by using details of building layouts, walking speed of users and a goal-driven user movement model. Analytical techniques are extremely complex and are not considered further in this paper.

along the boundary line. In other words, only enough information was recorded to answer the question "did the user enter zone A from zone B?"; the particular crossing position along a potentially very long border between zone A and zone B was not considered.

A more accurate model can be generated by arranging all the boundary lines in an arbitrary order to set up a one-dimensional coordinate system. In other words, the boundary lines can be arranged lengthwise along the real axis. A precise crossing point for the spatial domain can then be recorded as a single real value.

Recording the crossing point is significant because the better an attacker can model the users' movements from their observed past behaviour, the better he can predict what users do once they are hidden inside the mix zone. Therefore, being able to distinguish the position along the boundary line at which users cross into and out of the mix zone gives the attacker a better chance of matching entering and exiting users with increased reliability.

The middleware can record the ingress point i at time t and egress point e at time $t + \tau$ of user movement into and out of a mix zone and generate a movement matrix; this technique differs from the previous one because each row and column in the matrix representing boundary lines is replaced by discrete sections of boundary lines. There are three fundamental limits which prevent us from making the discrete boundary sections arbitrarily small:

Location accuracy The length of a discrete section cannot go below the accuracy of the underlying location system.

Sampling accuracy Intuitively, the greater the numbers of samples for each crossing section, the more accurate the estimate; therefore if the discretisation is too small, our estimates of user movement are likely to be inaccurate.[3]

Computational cost There is a computational cost associated with increasing the granularity of sampling; this is discussed further in the next section.

4. Determining the anonymity level

Users enter the mix zone with one pseudonym, change to a new unused pseudonym and, after an unknown length of time, exit under the new pseudonym.

The attacker can observe the times, coordinates and pseudonyms of all these ingress and egress events. His ideal goal is to reconstruct the correct mapping between all the ingress events and the egress events. This is equivalent to discovering the mapping between new and old pseudonyms.

How many such mappings are there? During the period of observation, assume there are n ingress events and n egress events.[4] The attacker observes n old pseudonyms going in, and n new pseudonyms coming out, most likely with some interleaving. Each permutation of the set of n new pseudonyms gives a new mapping, so there is a total of $n!$ mappings. Many of the mappings can be ruled out because:

- a user cannot exit a mix zone before they enter it,
- users cannot move between two non-connected mix zones without passing though an application zone (and therefore being sighted), and
- portions of boundary lines containing walls or other impassable objects prevent users entering or exiting at these locations.

These temporal and spatial restrictions are represented in the movement matrix as zero value cells and therefore the movement matrix is likely to be sparse.

The mapping problem faced by the attacker can be viewed as a weighted bipartite graph, where vertexes model ingress and egress pseudonyms and edge weights model the probability two pseudonyms represent the same underlying person. The edge weight probabilities can be estimated from the movement matrix by normalising the frequency count in each cell with the total number of movement sightings; edges with zero weight are removed from the graph. The *maximal cost perfect match*[5] of this bipartite graph represents the most probable mapping of incoming pseudonyms to outgoing ones.[6] Determining the maximal cost perfect match of a weighted bipartite graph has several polynomial time algorithms [5]. A simple algorithm to determine the maximal cost perfect match was developed by Kuhn [6] and is $O(ev^2)$ where e is the number of edges and v is the number of vertexes.

The maximal cost perfect match is the best result the attacker can produce; it represents the most

[3] More precisely, the central limit theorem states that, as the size of a random sample n increases, the distribution of the sample mean $\bar{X}$ tends towards $N(\mu, \sigma/\sqrt{n})$, where μ and σ are the mean and variance of the underlying population. Therefore increasing the number of samples, n, reduces the variance on our random sample mean, $\bar{X}$, and therefore $\bar{X}$ becomes a more accurate predictor of the underlying population mean.

[4] There are additional subtleties concerning uneven numbers of ingress and egress events (particularly concerning the use of any bipartite graph theory which follows). We shall ignore them for the moment, assuming a long term steady-state condition.

[5] A *perfect match* in a bipartite graph occurs when every vertex is connected to another vertex by a single edge. The perfect match with the highest summation of edge weights is the *maximal cost perfect match.*

[6] This assumes user behaviour is independent, and the likelihood of their movement can be accurately represented by the movement matrix.

likely de-anonymization of the underlying users passing through the mix zone. But this information alone is not as useful as it might sound: the attacker needs a measure of *confidence* in the quality of this result. Consider an example mix zone event with three possible mappings $M = \{m_1, m_2, m_3\}$ with the following probabilities $\{\frac{1}{100}, \frac{1}{150}, \frac{1}{150}\}$ respectively; knowing the most likely event $P(m_1) = \frac{1}{100}$ is not the whole story; what is really required is knowledge of how much more likely this mapping is when compared with the rest; *one* of these mappings must have occurred because these are the only mappings which explain this pattern of ingress and egress pseudonyms, at least according to the model. This conditional probability can be calculated as:

$$P(m_i|M) \overset{def}{=} \frac{P(m_i \wedge M)}{P(M)} = \frac{P(m_i)}{\sum_i P(m_i)} \qquad (1)$$

in this case because $m_i \subseteq M$.

The level of uncertainty in the set of possible mappings $m_i \in M$ can then be measured by using Shannon's classic entropy measure [7]:

$$h = -\sum_i P(m_i|M) log P(m_i|M) \qquad (2)$$

Intuitively, the entropy is related to the number of people you are indistinguishable from; if the entropy is b *bits* then 2^b users are indistinguishable from one another.

4.1. Computational and optimisation issues

The problem with this technique comes in calculating $P(m_i|M)$: the probabilities of all of the possible mappings must be calculated, and this is not computationally tractable because there are $n!$ of them.

Instead of calculating $P(M) = \sum_i P(m_i)$ directly, lazy evaluation can yield a lower bound $P_l(M) \leq P(M)$ by iterating through only perfect matchings in the bipartite graph.

Calculation of $P_l(M)$ in a lazy fashion starts by calculating the maximal cost perfect match (for example by using the Kuhn algorithm) and proceeds by searching for other perfect matchings. This can be done by finding an *alternating circuit* of edges with the following properties:

- each edge starts on the vertex the previous edge finished on,
- edges are alternately in the maximal cost perfect match and not, and
- no vertex appears in the alternating circuit more than once;

Therefore another perfect match can be generated from the set of edges in the alternating circuit and not in the maximal cost perfect match. Itai *et al* developed an algorithm based on this technique to iterate through all perfect matches one by one [8]. The worst case cost to find the next match (or determine no more matches exist) is $O(e)$.

The number of perfect matches cannot be precomputed, and there is no guaranteed bound on computation time before the last match is found. Therefore the lazy evaluation of $P_l(M)$ is a lower bound; its value cannot diminish because every new perfect match adds one term to denominator $\sum_i P(m_i)$ and probabilities are always in the range (0,1). Each time a new mapping is found the entropy or level of anonymity offered by the mix zone can be recalculated and can only go up. Therefore as lazy evaluation of $P_l(M)$ progresses one of three outcomes can occur:

- The level of anonymity in the mix zone rises to meet the minimum level desired by the users. The users have mixed sufficiently to thwart an attacker and the algorithm is terminated.
- The lazy evaluation terminates (i.e. all possible matches have been found), so $P_l(M) = P(M)$. If the level of anonymity in the mix zone is still not sufficient, the identities of the users could be compromised by an attacker.
- Computation time runs out (i.e. computation has gone on as long as practicable), therefore $P_l(M) \leq P(M)$. If the level of anonymity offered by the mix is still not sufficient it is unknown whether a sufficient level of anonymity will ever be reached for this mix (but, given similar computing power, the attacker is uncertain of the quality of his guess as well).

5. Individual user anonymity

So far we have discussed ways of measuring the level of anonymity experienced by users of the mix *collectively*. Curiously, for a particular user, the level of anonymity can change even *after* leaving the mix zone.

Consider the following simple scenario: a mix zone with four boundary lines, north (n), south (s), east (e) and west (w). To simplify the example we quantise the boundary very coarsely (e.g. all points on the north edge are treated as north) and assume any user entering the mix zone is guaranteed to have left after either one or two time periods. A movement matrix for this example is given in Table 1.

Consider the movements of two users u_1 and u_2 who enter the mix zone at the same time t, one from n and one from e respectively. If u_1 exits at time $t+1$ through

	n	s	e	w
n	$\frac{1}{64}$	$\frac{3}{64}$	$\frac{1}{32}$	$\frac{1}{32}$
s	$\frac{3}{64}$	$\frac{1}{64}$	$\frac{1}{32}$	$\frac{1}{32}$
e	$\frac{1}{32}$	$\frac{1}{32}$	$\frac{1}{64}$	$\frac{3}{64}$
w	$\frac{1}{32}$	$\frac{1}{32}$	$\frac{3}{64}$	$\frac{1}{64}$

Table 1. Movement matrix for $t+1$ and $t+2$.

s and u_2 remains in the mix zone, what level of mixing has occurred?

The measure of uncertainty is dependent on the exit taken by u_2 in the following time period $t+2$. If u_2 goes west, the possible mappings are $\{n \to s, e \to w\}$ and $\{e \to s, n \to w\}$. Of these, according to the probabilities encoded in the movement matrix, the first is much more likely. If, on the other hand u_2 goes south, then the possible mappings become $\{n \to s, e \to s\}$ and $\{e \to s, n \to s\}$, whose probabilities are identical. So, when u_2 goes south the attacker is much less certain about what happened and u_1 is much more anonymous (1 bit) than if u_2 had exited through west (0.47 bits). See Fig. 2 for a pictorial representation of user movements.

Given a movement matrix for a mix zone, the middleware can calculate a lower bound on the level of mixing for a particular user u (who is still in the mix zone) by assuming all the other current users leave the mix zone in the most probable manner. A lower bound on the level of mixing u experiences can then be calculated for each possible exit from the mix zone.

Intuitive user feedback is now possible, allowing the user to decide whether to suspend certain location-aware applications or take a detour if the level of privacy offered is too low. For example, the level of anonymity gained in a mix could be displayed as an "anonymity strength" readout on the location device (e.g. mobile phone).

6. Conclusions

In this paper we have refined the mix zone model, describing a quantifiable metric of location privacy from the point of view of the attacker. Analysis is computationally expensive and may require partial evaluation of the problem–we have described a method of achieving this. Furthermore, given fixed computational power there exists a trade-off between the tractability of the problem and the accuracy in which the real world is modelled.

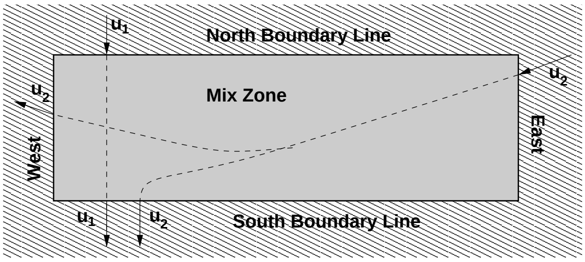


Figure 2. Entropy of the mix zone is dependent on the exit taken by u_2.

Evaluation of the level privacy gained before the user exits the mix zone provides increased usability and the possibility of providing continuous feedback to the user. In this scenario, the user can make decisions about whether to disable some location-aware services or to alter their movements in order to gain increased privacy.

Enabling location privacy is going to become increasingly important in a world where location-aware services are available over larger and larger geographical areas. Deploying systems that support location privacy for users and provide feedback about the level of anonymity users have may prove critical to the widespread adoption of location-aware services.

References

[1] J. R. Cuellar, J. B. Morris, D. K. Mulligan, J. Peterson, and J. Polk. Geopriv reqs. (IETF Internet draft), 2003.

[2] U. Hengartner and P. Steenkiste. Protecting access to people location information. In *Security in Pervasive Computing*, March 2003.

[3] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *ACM/USENIX International Conference on Mobile Systems, Applications and Services (MobiSys)*, 2003.

[4] A. R. Beresford and F. Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 3(1):46–55, 2003.

[5] Z. Galil. Efficient algorithms for finding maximum matching in graphs. *ACM Computing Surveys*, 18(1):23–38, March 1986.

[6] H. Kuhn. The hungarian method for the assignment problem. *Naval Res. Logist. Quart.*, 2, 1955.

[7] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.

[8] A. Itai, M. Rodeh, and S. Tanimoto. Some matching problems for bipartite graphs. *Journal of the Association for Computing Machinery*, 25(4):517–525, October 1978.