Location Privacy Protection Through Obfuscation

Jorge R Cuellar, Siemens AG, Corporate Technology*

December 2010

Introduction. Technical improvements of detection and transmission of location information, together with the widespread adoption of mobile communication devices and location-based services for business, social or informational purposes, are increasing the privacy concerns. Better, more elaborated solutions are necessary to provide users (*Rule Makers*, as they are called in *geopriv*) the possibility to express privacy policies and technical means to implement them. This is easier said than done.

Some important scenarios can be dealt with, in a satisfactory way using Pseudonym- or Anonymity-based techniques, because the real identity of the individual is irrelevant for the service. If necessary, the individual and the recipient of the location information can run a "pseudonym-agreement protocol", similar to some unauthenticated key agreements.

But there is a large class of situations in which the identity of the individual to be located is to be explicitly disclosed (or known) to the recipient. Let us suppose a user wants to make his position available via a presence service or within an online social forum to his contacts. Then, he will prefer to disclose his location with a high precision to close friends and with a low ones to others. In some cases, he may wish to disclose his location within 10 or within 100 km. Further, let us assume, he does not want to lie about his location, and the algorithm should not force him to do so.

Problems with current solutions. For a *single location provision*, obfuscation (understood as degrading the quality of information about a user's location) renders an easy and unproblematic solution. A "cloaked location" is essentially a region that contains the user's true location.

Even though there is a reasonable amount of literature discussing cloacking and obfuscation procedures, there is no consensus on how to solve the following frequent problems:

1. The combination of an obscured location with public geographic information (highways, lakes, mountains, cities, etc) may render a much precise location information than desired. If you are traveling roughly west at 110km/hr in a certain region, one may easily guess where you are.
2. The necessity of updating location information due to movement may disclose small areas during the updates.
3. The fact that people travel, work or live together and their policies may interfere, can be used to obtain the location of a user. If my daughter publishes *her* location with a high precision, she will also be disclosing where *I* live, if the location recipient knows that we live together.
4. The fact that people visit the same locations regularly can be exploited to obtain those locations with high precision.

*This work has been supported by the European Community's Seventh Framework Programme under the project NESSoS, Project no. 256980.

The solution to the first problem could involve advising the user that under the current conditions the information leakage is higher than expected and changing automatically or interactively the parameters of the policy.

The second problem is not too difficult to solve: it is possible to change the obscured location provided before being it absolutely necessary. But this may conflict with solutions to other problems mentioned.

The third problem will require a long effort to solve. Under circumstances that must be specified, not only the policies of several people should be simultaneously satisfied, but also the obfuscation algorithms of the different people have to deliver the same result.

The fourth problem requires a standardized, common solution, as discussed now. This is our main point in this position paper.

Why Location Obfuscation MUST be standardized Suppose John goes home every night. If the reported obfuscated locations are all different, an analysis will probably soon reveal the home location with high precision. Also, to reduce the amount of information leaked, the reported location may not change abruptly with small variations of the real position, as this one is measured with some imprecision or the target may move a bit. And also the *shift* (vectorial difference between average of the centroid of reported location and real location) can not be a continuous invertible function (else, since the shift is leaked, the location can be calculated).

Or suppose John visits Berlin regularly, every Thursday. One of his contacts conjectures he could be visiting one of 2 companies. If he obtains regularly John's obfuscated locations, and they are always different, clustering algorithms (and the intersection of the provided locations) will eventually probably disclose evidence about which company John is regularly visiting. We say that the two locations are *distinguishable* via the algorithm.

Being an equivalence relationship, indistinguishability partitions the space into "*i-blocks*", or *indistinguishability blocks*. That is: two points are in the same i-block if and only if they are indistinguishable. *Any* obscuring algorithm partitions the space into i-blocks. The larger the i-blocks, the less the algorithm leaks. If A_{prov} is the area of the provided location, and A_{block} is the area of the i-blocks, $1 - A_{block}/A_{prov}$ is a measure of the relative *leakage* of the algorithm. If any two points are distinguishable, i-blocks have area 0 and the leakage is 1.

One solution is to explicitly construct *blocks* as big as possible and to make the algorithm depend on the block, not on the precise point within the block.

Several ways of generating convenient families of *blocks* can be designed, some based on map projections and geometric grids (say, given by latitude, longitude & altitude) and some based on countries, cities, landmarks (using Voronoi tessellations), etc. To alleviate the problem 2 above, it is necessary to introduce *transition blocks* between other blocks, in order to diffuse the area when moving from one block to another.

Whatever method is chosen, it MUST be standardized in detail, because the results of runs of *different* obscuring algorithms will otherwise provide a high information leakage and will disclose the places that users visit regularly.

One particularly simple version of the algorithm constructing i-blocks based on a rectangular grid is given in draft-ietf-geopriv-policy-22

Some Open Questions

- The notion of *leakage* mentioned does not consider "typical use cases", number of runs of the algorithm, probability of leakage, etc. Which variants are useful?
- What algorithms are optimal (leak the least), given one definition of leakage?
- What user preferences or history values can be used for the output of the algorithm?
- What geometric grids can be used for very large regions, or the entire globe?