*Ninth LACCEI Latin American and Caribbean Conference (LACCEI'2011), Engineering for a Smart Planet, Innovation, Information Technology and Computational Tools for Sustainable Development, August 3-5, 2011, Medellín, Colombia.*

# Secure location-based service for social networks

Carolina Marin
Dept. of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431,
cmarin8@fau.edu

Eduardo B. Fernandez
Dept. of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431,
ed@cse.fau.edu

Maria M. Larrondo Petrie
Dept. of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431,
petrie@fau.edu

## Abstract

Nowadays, the rapid growth of social networks using mobile personal devices has resulted in the use of location-based services for different purposes. Applications such as Geo-tagging are now very popular in smart phones. However, current models for exchange of this information may require users to yield some of their privacy and security unless we use secure ways to protect their location information. We present here a pattern that describes how a user can find his friends using location-based services in a secure way.

## 1. Introduction

Social network sites attract millions of people on the internet and many of these people have integrated these sites into their daily life. The widespread use of cellular telephones and the availability of user location information are facilitating personalized location-based applications. They are part of a new suite of emerging social networking tools that run on the Web 2.0 platform. Location-based services offer users the ability to look up the location of a friend using a smart phone, desktop, or other device, anytime and anywhere.

Mobile devices with geo-positioning capabilities are becoming cheaper and more popular. By sharing their location information (e.g., via Wi-Fi, Bluetooth, or GPRS), mobile users have available a variety of location based services. An interesting type of such services is a friend-locator service, which shows users their friends' locations on a map and helps identify nearby friends. Friend locators together with other mobile social networking services are predicted to become a multi-billion dollar industry over the next few years. Several friend-locator services, such as iPoki, Google Latitude, and Fire Eagle are already available. Also, social networks sites such as Facebook, Twitter and Flickr have applications that use location-based services (Brooker, D. et al., 2010).

Although, location-based services are popular among social network users there are several privacy and security problems that should be considered. Strangers can get this information easily through the social network platform due to their lack of security. To use location-based services users should define what they want to share and what they want to hide, and their wishes should be enforced by the system (Yiu, M. L. et al., 2010).

We present here a pattern that allows people to know the location of other people (friends), who have agreed on this contact. In Section 2 we describe our pattern, and Section 3 presents some conclusions and indicates our future work.

## 2. THE SECURE LOCATION-BASED SERVICE PATTERN

### Intent

Provide a person with the location of those friends who have allowed their locations to be shown and restrict the location information only to this group. The people in the group may also temporarily block their locations.

### Example

Alice and several of her friends have agreed on sharing their locations, this is convenient for their socializing. However, occasionally strangers find out their locations. This is a privacy violation and may endanger them

### Context

This pattern can be used in social networks where location services are available.

### Problem

We want to know the location of some of our friends and we want them to know where we are. However, we don’t want strangers to find out this information.

The solution for this problem is affected by the following **forces**:

- *Location*: it may be convenient and useful to know where your friends are and that they know where you are.
- *Security*: if location-based services are not protected, strangers may find out where you are in what city you live, and even your address. This may violate your privacy and maybe endanger you.
- *Privacy*: if strangers can find your location, this is a privacy violation.
- *Control*: people may want to hide their visibility at some times. There should be an easy way to do this.

### Solution

Define a circle of friends who want to be in contact with each other and want to know their locations at all times except when they block their visibility, we need to control access to their locations.

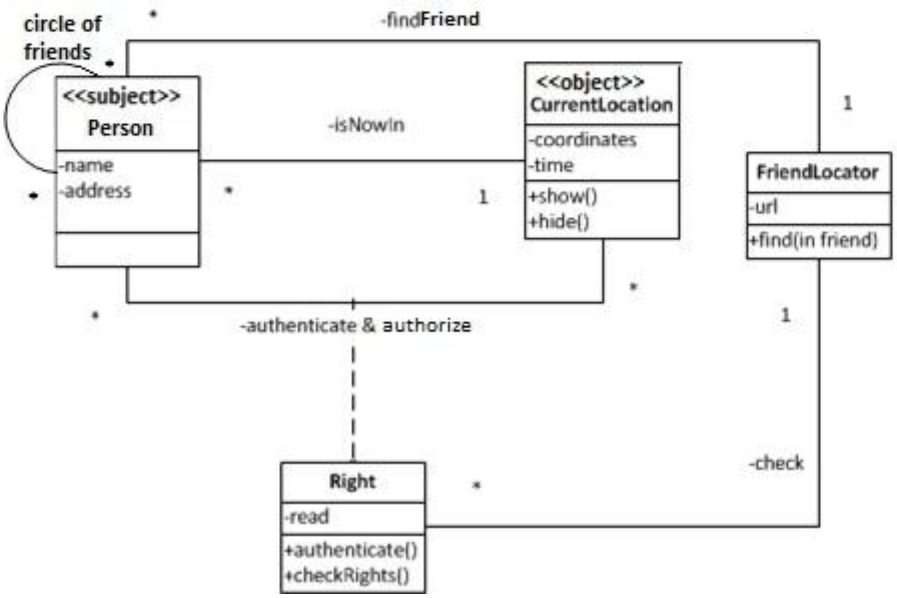

**Figure 1: Class diagram for the Secure Location-Based Service pattern**

### Structure

A **Person** is part of a *circle of friends*, who trust each other. A **Person** is in some **CurrentLocation** at a given time. Only the other friends in the circle can see (read) his location. A person can hide her location whenever he wants. If the requester is authorized, the **FriendLocator** will find the location of the requested person. Note that we are applying here the Authorization pattern, where subject and object stereotypes define those 2 elements of the model; a person can show or hide his location according to his wishes.

### *Dynamics*

The class diagram of Figure 1 supports the use cases shown in Figure 2. Both use cases require user authentication Find Friend also requires access control.

The sequence diagram of Figure 3 shows the use case "Find friend". In this figure friend1 tries to find friend2. The FriendLocator checks to see if friend1 is authorized to see this location. If "yes", it asks for friend2's location to Current Location. An object in this class returns the current location of friend2 to friend1. This sequence diagram assumes friend1 has already been authenticated.

### Implementation

- Social networks with location-based services like Facebook can be used to match users with a place, event or local group to socialize in or enable a group of users to decide on a meeting activity.
- Several mobile operating systems, e.g. Symbian and Linux, include location services as part of their application libraries.

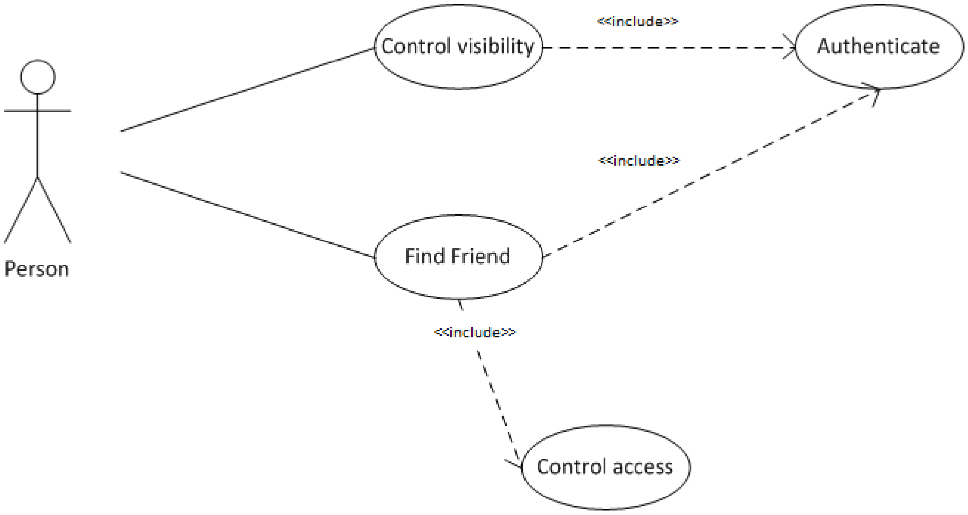

**Figure 2: Use Case Diagram for find friend**

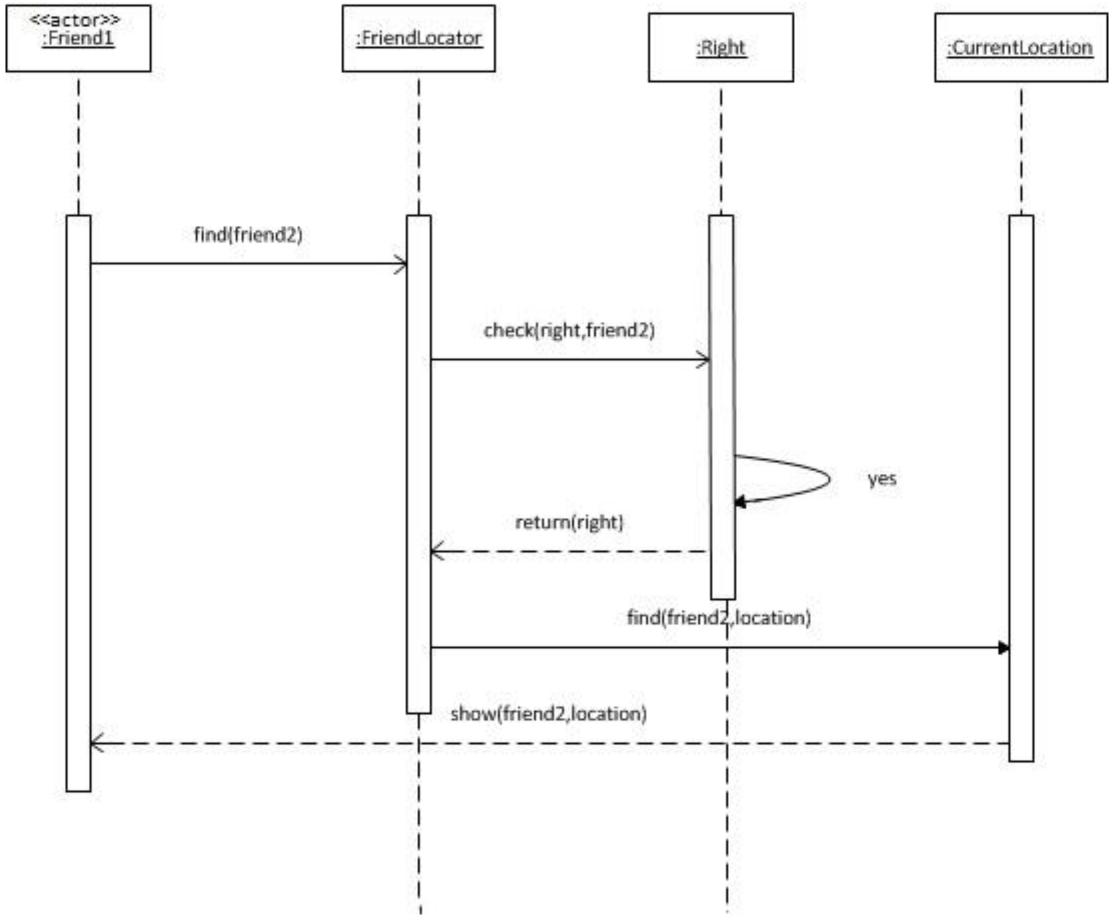

**Figure 3: Sequence Diagram for the use case find friend**

**Known Uses**

- **IPoki**: a GPS-based social network that allows people to connect with others to share geolocation information. With Ipoki the user can change his privacy settings at any time to control who can see what (Ipoki 2011).

- **Google Latitude**: a location-aware mobile app developed by Google. Latitude allows mobile phone users to allow specific people to view their current location. Via their own Google Account, the user's cell phone location is mapped on Google Maps. The user can control the accuracy and details of what each of the other users can see — an exact location can be allowed, or it can be limited to identifying the city only. For privacy, it can also be turned off by the user, or a location can be manually entered. Users have to explicitly opt in to Latitude, and may only see the location of those friends who have decided to share their location with them (Wikipedia 2011).

- **Fire Eagle:** a Yahoo! owned service that acts as a store for a user's location. A user can authorize other services and applications to update or access this information via the Fire Eagle API, allowing a user to update their location once and then use it on any Fire Eagle enabled-website (Wikipedia 2011).

- **Waze**: a GPS application featuring turn-by-turn navigation, developed by Waze Mobile for mobile phones. Waze is a community-driven application and learns from users' driving times to provide routing and real-time traffic updates. It is free to download and use, as it gathers map data and other information from users who use the service. Additionally, people can report accidents, traffic jams, speed traps, and police and can update roads, landmarks, house numbers, etc. (Wikipedia 2011).

### Consequences

This pattern has the following advantages:

- *Location*: a person can find the location of his friends in the circle of friends.
- *Security*: the Friend Locator can restrict access to only the friends in the circle.
- *Control*: a person can hide his location at certain times.
- *Privacy:* no strangers can find out the location of a person.

The pattern has the following liabilities:

- We don't get announcements if a friend is close to our location, we need to ask for locations. The model can be extended to include this feature.
- We can only belong to one circle of friends. The model can be extended to include this feature.

### Related Patterns

- The Circle of Trust pattern: allow the formation of trust relationships among service providers in order for their subjects to access an integrated and more secure environment (Delessy, N. et al., 2007).
- Reference Monitor: enforce authorizations when a subject requests a protection object and provide the subject with a decision (Schumacher, M. et al., 2006).
- Authorization: this pattern describes who is authorized to access specific resources in a system, in an environment in which we have resources whose access needs to be controlled. It indicates, for each active entity that can access resources, which resources it can access, and how it can access them (Fernandez, E. B. et al., 2008).

## CONCLUSIONS

We are building a catalog of patterns intended to help developers build secure social networks. Our previous patterns include [Marin et al. 2010]:

- The Participation and Collaboration pattern describes the functionality of the collaboration between users participating in social networks, together with access and rights restrictions.
- The Collaborative Tagging pattern makes content more meaningful and useful by using keywords to tag bookmarks, photographs, and other content.

This paper complements those patterns by adding another aspect of increasing value for social networks. We will extend this pattern with the features mentioned in the Consequences section. Our next pattern will describe how to build secure mashups.

## REFERENCES

Brooker, D., Carey,T., Warren, I. (2010). Middleware for Social Networking on Mobile Devices, Proc of the 21 Australian Software Engineering Conference.

Delessy, N., Fernandez ,E.B, Larrondo-Petrie M.M. (2007). "A pattern language for identity management", Procs. of the 2nd IEEE Int. Multiconference on Computing in the Global Information, March 4-9, Guadeloupe, French Caribbean.

Fernandez, E. B.,Gudes, E., Olivier, M.(2005). The design of secure systems. Chapter 8.

Fire Eagle (03/02/11) http://en.wikipedia.org/wiki/Fire_Eagle

Google Latitude (03/02/11) http://en.wikipedia.org/wiki/Google_Latitude

Ipoki (03/02/11) http://www.ipoki.com/

Marin, C., Fernandez, E.B., Larrondo-Petrie, M.M., "Patterns for social networks in Web 2.0", *Procs. of the 8th Latin American and Caribbean Conf. for Eng. and Technology (LACCEI'2010),* June 1-4, 2010, Arequipa, Peru.

Schumacher, M.,Fernandez, E.B., Hybertson, D.,Buschmann, F.,Sommerlad, P.(2006) Security Patterns: Integrating security and systems engineering",Wiley Series on Software Design Patterns.

Waze (03/02/11) http://en.wikipedia.org/wiki/Waze

Yiu, M L., Šaltenis, S., Šikšny, L., Thomsen, J. P.(2010). Private and Flexible Proximity Detection in Mobile Social Networks Eleventh International Conference on Mobile Data Management.

### *Authorization and Disclaimer*

*Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.*