Protecting People Location Information (Extended Abstract)

Urs Hengartner[1] and Peter Steenkiste[1,2]

[1] Department of Computer Science
[2] Department of Electrical and Computer Engineering
Carnegie Mellon University
{uhengart,prs}@cs.cmu.edu

1 Introduction

Ubiquitous computing environments rely on the availability of location information about people to provide location-specific services. However, location is a sensitive piece of information. For example, it is often possible to gather information about the current activities of a person from her location. Therefore, location information should not be distributed to just anyone. The solution is to allow people to specify location policies that state who should be allowed to locate them. In this paper, we present and motivate key features that should be supported by location policies.

We have implemented a prototype of a secure people location service that supports many of the features discussed here, and we are currently evaluating the system with real users in the context of the CMU Aura project [1].

2 Location Policies

We discuss desirable features of location policies.

2.1 User and Room Location Policies

Location queries can ask either for the location of a user ("user query") or for the people in or at a geographical location, such as a room in a building ("room query"). Thus we need two types of location policies: user policies and room polices. A user policy states who is allowed to get location information about a user. For example, "Bob is allowed to find out about Alice's location". Similarly, a room policy specifies who is allowed to find out about the people currently in a room. For example, "Bob is allowed to find out about the people in Alice's office".

However, location policies should not be restricted to all-or-nothing access control. We believe location policies should provide fine grain control for at least the following properties:

Granularity. A policy should be able to restrict the granularity of the returned location information. For example, a user policy can state that only a building name be returned instead of an actual room (e.g., "CMU Wean Hall" vs. "CMU Wean Hall

8220"). A room location policy can require that the number of people in a room is returned instead of the identity of the people in the room (e.g., "two people" vs. "Alice and Bob").

Locations/users. User policies can limit location information to a set of locations, so location information will be returned only if the queried user is at one of the listed locations. For example, "Bob is allowed to find out about Alice's location only if Alice is in her office". Similarly, room location policies can be limited to a set of users, so a room query will include only users listed in the policy (provided they are in the room).

Time intervals. Policies can limit time intervals during which access should be granted. For example, access can be restricted to working hours only.

2.2 Transitivity of Access Rights

When Bob is granted access to Alice's location information, should Bob be allowed to forward this access right to Carol? Similarly, when Ed is allowed to learn about the people in his office, should he be allowed to grant this privilege to Fred? In short, should access rights to location information be transitive?

There is no simple universal answer to this question. For the first case, it should probably be Alice's decision whether this forwarding should be allowed. For the second case, we argue that in most cases, the answer should be no. If Ed is given the option to let other people find out about Alice's location when she is in his office, Alice's location policy might be violated. In addition, Ed could give someone not within the same institution access rights to his office, which might not be in the institution's interest.

Based on this discussion, we argue that people that define policies should be able to explicitly state whether they want access rights to be transitive. In some cases, the decision on whether access rights should be transitive depends on the environment. We examine some example environments in Section 3.

2.3 Conflicting Policies

User and room location policies can be conflicting. For example, assume that Alice does not allow Bob to locate her, but Carol allows Bob to locate people in her office. If Alice is in Carol's office, should Bob be able to learn about this fact? There are three possible ways for dealing with this issue:

- The room policy is ignored when answering user queries. Similarly, the user policy is ignored for room queries. In our example, Bob would thus see Alice being in Carol's office.
- Both policies are looked at for any request and information is returned only if it is approved by both of them. Bob would thus not see Alice being in Carol's office.
- The user and room policies are established in a synchronized fashion so that no conflicts arise. Alice and Carol's location policies would thus have to be rewritten. As an example of this approach, Leonhardt and Magee [2] suggest authorizing user/room pairs. A drawback of such scheme is the potentially huge number of pairs for which a policy needs to be established.

What approach is most appropriate depends on the environment in which people are being located. For example, earlier work (e.g., by Spreitzer and Theimer [4]) lets users have influence on room policies. One example is that users can specify whether room queries for a room will include them in their answer, effectively giving user policies precedence over room policies. While this approach is appropriate for some scenarios (e.g. public buildings), it is not for others. For example, once can argue that in general, the owner of a room should always be able to find out who is in her room, regardless of the user policies.

2.4 User vs. Institutional Policies

Depending on the environment, location policies are specified by different entities. For some environments, a central authority will define these policies, whereas for other ones, users may be allowed to specify some of the policies, e.g. who should be allowed to locate them.

Environments may not only differ in who can specify policies, but also in how policies are applied. We saw two examples of this already: transitivity of policies and the resolution of conflicts between user and room location policies.

3 Example Environments

An access control system for people location information should be flexible enough so it can be configured for different environments. To illustrate why flexibility is needed, we discuss how location policies would be set and deployed in two different environments: a hospital and a university environment.

3.1 Hospital

Medical data is typically protected based on a multilateral security model that protects information flow between compartments. For example, only doctors taking care of a patient have access to her medical data, but not every doctor in the hospital. For location information, a similar model should be applied. A patient should be locatable only by her doctors. In addition, a patient should be able to allow other people (e.g., her husband) to locate her. To ensure this policy, a central authority has to take care of establishing policies and the patient should be given the right to include additional people in her location policy.

Room policies should be established by the central authority to protect the patient's privacy. User and room location policies do not need to be synchronized for the hospital scenario.

3.2 University

In a university setting, university members are allowed to specify their user policies. However, room policies are established by a central authority. For offices, the authority

is likely to delegate the right to establish the room policy to the occupant of the office. For lecture rooms and hallways, the authority would typically set the room policy such that the user policies of users in the room/hallway are contacted upon a query. Therefore, user and room location policies become synchronized.

User policies might be transitive. For room policies, the institution might decide to not let the occupant of an office redelegate his rights to other people.

4 Conclusion and Status

In this paper, we have analyzed some requirements of location policies. We have implemented an access control system that supports flexible location policies as part of a people location system that is currently being deployed at Carnegie Mellon. The people location system uses a variety of sources of information to locate people, e.g. wireless network data [3], calendar information, and login information.

Our system uses digital certificates for expressing location policies. Digital certificates offer a high degree of flexibility and allow the formulation of complex location policies required to support the various constraints outlined in this paper. Though in our environment users can specify their own policies by issuing appropriate certificates through a user interface application, it is possible to have certificates issued by a central authority. Therefore, the same basic mechanisms can be used in different environments.

We are currently evaluating our people location system with real users. The evaluation will give us a better picture of the kinds of location policies that people specify and of the constraints that they set in their location policies.

References

1. David Garlan, Daniel Siewiorek, Asim Smailagic, and Peter Steenkiste. Project Aura: Towards Distraction-Free Pervasive Computing. *IEEE Pervasive Computing*, 1(2):22–31, April-June 2002.
2. U. Leonhardt and J. Magee. Security Considerations for a Distributed Location Service. *Journal of Network and Systems Management*, 6(1):51–70, March 1998.
3. A. Smailagic, D. Siewiorek, J. Anhalt, and F. Gemperle. Towards context aware computing. *IEEE Intelligent Systems*, 6(3):38–46, June 2001.
4. M. Spreitzer and M. Theimer. Providing Location Information in a Ubiquitous Computing Environment. In *Proceedings of SIGOPS '93*, pages 270–283, Dec 1993.