Privacy management for social awareness applications

Mika Raento[*,†], Antti Oulasvirta[*]

Helsinki Institute for Information Technology[*]
Department of Computer Science, University of Helsinki[†]
{mraento,oulasvir}@cs.helsinki.fi

Abstract. We analyze how the social psychological theories of self-disclosure and privacy as boundary negotiation can be applied to ubiquitous social applications. A conceptual framework of privacy management is derived from that theory, the framework is used to derive system requirements and they are evaluated in the context of a concrete social awareness service.

1 Introduction

We have been building and testing an ubiquitous social awareness service, ContextContacts[1]. Social awareness applications are designed to re-contextualize remote and mediated communication and tasks by transmitting, automatically or in a user-controlled manner, cues of people's current state or situation. The goal of such systems is to build "an understanding of the activities of others which provides a context of your own activity" [2, p. 107].

Ubiquitous computing is meant to participate in the everyday life of people, as opposed to restricted spatial, temporal and social settings, so any ubiquitous service has to deal with privacy [3]. A social awareness service that is *focused* on handling personal data doubly so.

In this paper we present the idea of privacy *management* and its constituent concepts like control, transparency, reciprocity and accountability. These concepts are grounded in the analysis of privacy in the fields of ethics, social psychology, law and computing. Our contribution is in applying and elaborating existing theory of self-disclosure to the design of ubiquitous computer systems in the scope of individual-to-individual interaction (as opposed to interaction between individuals and organizations).

The design requirements are discussed through ContextContacts and the IETF XMPP [4] and SIMPLE [5] presence protocols. XMPP's predecessor Jabber was the first open instant messaging protocol and is very widely used on the Internet. SIMPLE has been adopted for 3G telephony and is used in upcoming enterprise messaging applications. Both will most certainly continue to play a major role in presence services.

2 Privacy management

Privacy means different things in different contexts, in different cultures and to different people. We are mostly interested in a descriptive, instead of normative, definition. We shall view privacy via social psychologist Irwin Altman's characterization of privacy as a dynamic process of negotiating the boundary between the individual and the environment[6]. This is quite different from the normative views found in legislation or guidelines.[7]

The different views of privacy stem from focusing on different social scopes: privacy can relate to the boundary between the individual and the general public, individual and an organization or other individuals. Social awareness applications, like ContextContacts, deal mostly in the realm of privacy between individuals, which shall be our focus. If introduced into organizational settings they will have to take into account also the individual–organization boundary.

We feel that social awareness services must enable privacy *management* — the continuous social *process* of negotiating the self–environment boundary [6]. Although this idea is not ours or completely new, it has not been as widely used in computing as one could imagine. Instead many publications talk about privacy *preservation* (witness 420 hits on Google Scholar with the keyword "privacy-preserving"), as if privacy was a state of the world that is on or off. This view on privacy is due to the focus of computer science on the larger scopes.

Grounding our ideas mainly in social psychology, but drawing inspiration also from philosophy, political science, law, economics and computer science, we dismantle privacy management into the following constituents:

Control Type and extent of information revealed to others is decided dynamically according to situationally arising needs and demands.

Accountability The discloser of information perceives breaches of implicit or explicit agreements on using revealed information and holds the other person accountable for those actions.

Plausible deniability When being asked about something private, a person can plausibly deny noticing or understanding the question instead of appearing to refuse to answer.

Reciprocity The disclosure is normally not one-sided, but mostly symmetrical — the amount of disclosure from A to B is strongly related to the amount of disclosure from B to A.

Utility Can the utility of private data be measured, can it be measured by individuals and can it be traded?

These concepts are not independent, orthogonal, non-contradictory, or constitute the totality of privacy management. We feel that they are the most important, most central and most informative. The concepts will be concretized by showing how they have influenced and will influence the design of ContextContacts (an example of information revealed to others shown in Fig. 1), how the IETF standard presence protocols XMPP and SIMPLE support them, and what changes to the standards may be necessary.

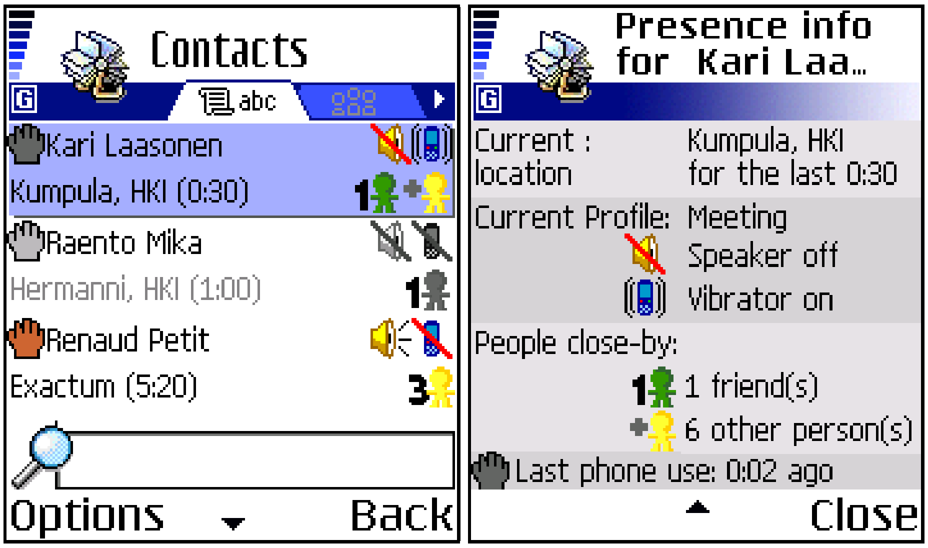

Fig. 1. ContextContacts. Mobile phone Contacts augmented with information on the current and past location, phone usage activity, people present and phone profile.

One notable abstention from this list is the concept of anonymity (or even pseudonymity). The requirements we focus on are meant to model interactions between participants, not of participants and third parties. The whole idea is to be able to identify the persons related to by the presence information. This does not mean that anonymity is not a requirement from other perspectives.

3 Control

The most important aspect of privacy management is control. Control is however not absolute, as privacy is a process of *negotiation* [6] — neither side in a negotiation can dictate the outcome, but both sides must have control over their decisions. People want to control the when, the what and the who of disclosure.

Consolvo et al. [8] show convincingly how dynamic the nature of mediated location disclosure is. They use experience-sampling to probe users' willingness to share their current location with their peers in real-life situations. They show that although the most important factor in disclosure is the identity of the asker or observer, there are no static rules that can decide what is revealed but that is instead completely situation-dependant.

Control in a computer system is no simple thing. In real-world human communication self-disclosure is managed (mostly) intuitively, unconsciously and automatically [9]. A computer system is not, at least not before being used extensively, part of the unconscious understanding of users. The users have to be able to build a correct-enough mental model of the system in relation to their actions, they have to be capable of carrying out the necessary actions and willing to carry out those actions. Lederer et al. [10] identify five pitfalls in designing control mechanisms for disclosure in ubiquitous computing:

1. Obscuring potential information flow. Users have to know and understand what *might* be revealed. Otherwise they may feel betrayed if more is revealed than they thought, or unnecessarily anxious when less.
2. Obscuring actual information flow. Who got to see what? This is elaborated in the section on Accountability.
3. Emphasizing conguration over action. Static configuration is not enough for privacy management, users are not motivated to spend a lot of time just to take the system into use and they do not understand privacy settings that are not situated into actual use.
4. Lacking coarse-grained control. The management of disclosure and privacy has to be effortless, or as nearly effortless as possible.
5. Inhibiting established practice. Since managing technically mediated disclosure is more difficult anyway, we should at least support the known *practices* of control rather than requiring users to construct new ones.

Point 4. above presents a difficult interaction-design problem. The social psychological research shows that not only is the control of self-disclosure highly granular but it is complicated by the feature of *relational selves* [11]:users may have nearly as many selves as they have significant interpersonal relationships. These facts argue for having a fine-grained and complex control interface when revealing information through a computer system. On the other hand the interface has to be very simple to be easy to use and efficient, as cognitive resources are extremely scarce in mobile use [12]. Making control coarse is one way to make it simple, iterative and user-ccentered user-interfae optimization another.

Leysia Palen showed [13] showed both that people seemingly stick to default settings even with very personal data and that they do find ways to dynamically control disclosure. Palen studied shared calendaring systems at both Sun Microsystems and Microsoft. At Microsoft by default the calendaring system only showed free/busy information whereas at Sun the complete description of events were shown. At both, approximately 90% of users never changed the visibility setting. But, users at Sun were aware of the disclosure and managed it by entering cryptic descriptions for private events.

As a necessary condition for control, ContextContacts allows users to build an understanding of the potential information flow. The *self-view* shows at any time the information currently sent to others, in exactly the same way as others' presence is shown. The first version of ContextContacts has only the coarse grained control: it allows the user to switch the awareness service off, fairly easily (the on/off setting was the first choice in the application menu, but users had to know how to switch to this background application). The next version will enable both rapidly switching visibility on/off but also to control what is shown to which groups of individuals dynamically, as shown in Fig.2. It is very hard to say in advance whether users will use the finer-grained control.

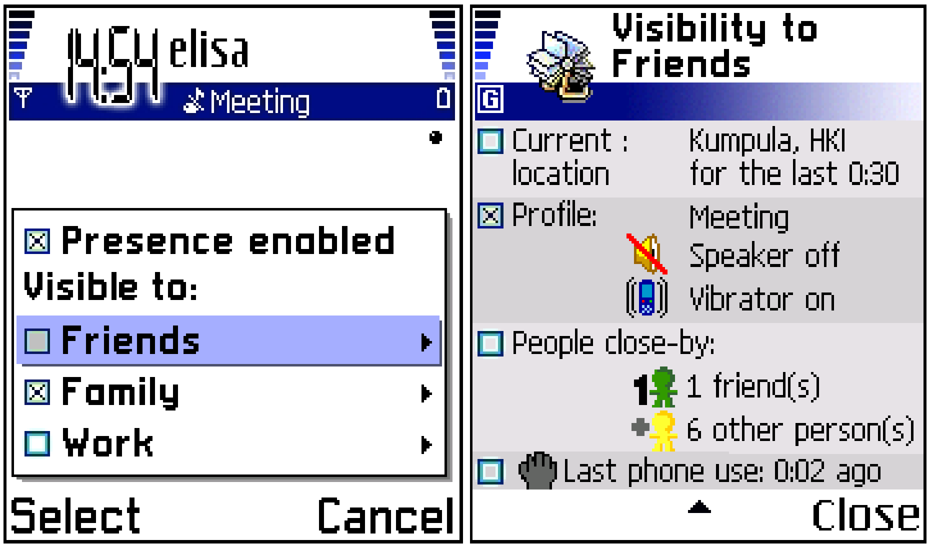

Fig. 2. Visibility control. The user can quickly switch on and off the awareness service, hide themself from a contact group or control in detail what information is revealed.

The Jabber/XMPP presence protocol allows users to stop the flow of presence information and to select which individuals have access to the information. Presence information may be sent only to some individuals, although the specification does emphasize that sending to all subscribers is the normal course of action. The protocol itself doesn't necessitate disclosure of all known presence information, so it does support control of what is revealed. XMPP should support giving different views of oneself to different observers in a clearer manner, and the semantics of targeted and general presence announcements could be clarified (e.g., how does the automatic notifications generated by connections and disconnections relate to previously sent directed presence?). SIMPLE has an extensive mechanism for specifying access rules.

4 Accountability

With unmediated self-disclosure people have an intuitive awareness of how much others get to know about them. They do not have to guess who might have access to the information, what they are revealing and what might happen with it. The recipient of the disclosure is accountable for use of that information.

In proposing a social awareness service we change the setting in two ways: instead of the individual actively disclosing information the information is *collected* and even translated automatically, and it is *transmitted* automatically without conscious action from the individual. There is now no automatic knowledge of what has been revealed to base accountability on. To bring back that knowledge the system has to track who has seen what and make a log of those revelations accessible to the discloser. By making the observation visible, it should not be felt as surveillance. We should avoid the shadow of Big Brother.

Fig. 3. Lookup Log. Time and date of presence lookups made by others are shown, and by clicking on the log item the information that was revealed is shown.

The next version of ContextContacts will track “lookups” of presence. This requires such lookups to be observable. Instead of showing the presence directly in the Contacts (Fig.1), the user will need to click once to see it. The lookups will be communicated with audio-tactile feedback to the observed, and a log of them will be made available through the familiar channel of logs of received and made calls, as shown in Fig.3. The accountability clearly is a tradeoff in terms of usability: instead of presence information being immediately available when making the call, another interaction step is needed requiring a longer time. On the other hand, as the lookup is made observable, it becomes a communicational resource to the users: by looking somebody up you can communicate your interest or need to speak with them in an effortless and embodied manner.

Neither XMPP or SIMPLE provide for this kind of accountability, although they are used to carry increasingly private informaiton. These protocols should be changed to support, if not mandate, pull based notification and tracking of such pulls (client-side accountability implementations are vulnerable to malicious clients, and credit-based tracking doesn’t tell when the information was actually used).

5 Plausible deniability

Lederer et al. argue that the ability to plausible deny refusing to disclose something is central to the control aspect of privacy management [10]. Aoki and Woordruff argue that if such refusals can be made *ambiguous*, they are not as detrimental to the social relationship [14].

Deniability is a clear component of mediated disclosure. Teenagers do not always answer their mobile phones, claiming that they didn't hear the ringing or that their battery was dead [15]. It happens in unmediated disclosure as well: if you are asked about something you don't want to reveal, you may pretend to misunderstand, change the subject or answer vaguely. The plausibility is in the other believing that you did mishear or accepting the vague answer.

Plausible deniability is clearly problematic for ContextContacts. It is supposed to be always on and automatic, there is no human aspect to be the point of deniability. At the moment the service cannot transmit data when a phone call is in progress, the location is vague and GPRS coverage intermittent. The more efficient the system becomes, the less ambiguous are omissions.

Since it is hard to attribute denial of information to the system, an alternative approach has to be used. We will implement a way to *fake* some of the presence information in ContextContacts, the same way many instant messaging applications allow the user's presence to be set manually. There are many possible approaches to such faking: randomly generating data, mimicing 'typical' data or manually inputting values. We shall first test a simple way of manual specification: freezing the values sent to others to the current values.

If the support for showing different views to different people is added to XMPP, as detailed in the Control section, it would be enough for deniability as well. The protocol must make sure that there is no way for the recipient to distinguish between real and faked presence information. The SIMPLE access control mechanism does not provide for giving different data to different subscribers, just different amounts of data. Both would benefit from explicit support for per-subscriber views of, say, location.

6 Reciprocity

Studies of self-disclosure have revealed that reciprocality in self- disclosure between the partners is necessary for building of trust and deepening of relationship [16] and people who do not tell of themselves or tell too much are generally disliked [9] (the opposite is not necessarily true: disclosure does not automatically make someone likable [17]). Although participants in social interaction do not tell exactly the same things about themselves, there is a very strong sense of balance in the amounts and kinds revealed.

It has been assumed by us and others [18,19] that reciprocity should be mandated by the service to be effective. In the current version of ContextContacts a technical form reciprocity is easily achieved. Since the only control off visibility is completely on or completely off, reciprocity means showing others only when the user themself is visible. This is a simplistic version of reciprocity: it only balances *time*: I'm able to see you for the same amount of time as you can see me. If we have different levels of activity or do different things we reveal different amounts of ourselves. On the other hand it is quite straightforward and understandable.

The addition of more fine-grained control over what information is revealed makes reciprocity more complicated. If A doesn't reveal location to B, should B's location automatically be hidden from A? What if B *wants* to reveal their location? How much does reciprocity interfere with control in that case? Should B's information be faked in some way if A is faking towards B? Will the users have any chance of building an accurate mental model if such complex automation of reciprocity are implemented? We are not quite decided yet, but will probably implement a mechanism whereby A won't see those aspects of B that they are not willing to reveal themself. Only experiments will tell whether this is workable.

XMPP and SIMPLE protocols are agnostic as to the concept of reciprocity. Subscriptions and presence notifications can be as symmetrical or asymmetrical as needed. This is probably the right choice: client applications and users are free to build the level of reciprocity that is needed for a specific application, and by being agnostic the protocol enables many different patterns of usage.

7 Utility

Libertarian economists may want us to believe that privacy is just another commodity be traded and that a free market will result in an equilibrium in regard to privacy and benefit. This view is reflected in the stakeholder comments to the US Federal Trade Commission's report on Online Access and Security [20]. As utility is the driving force behind most technical designs, presenting the limits of designing privacy management based on utility is important.

In general current economic theory identifies several problems with human action in a marketplace: Future and projection discounting, people tend to give disproportionately large valuations to immediate returns [21] and assume that current preferences will hold in the future [22]. Subjectivity of utility, based on values and beliefs. Incomplete information and bounded rationality. [23]

Information is even more incomplete with computer systems. The capabilities of future information systems are *highly* unpredictable: E.g., nobody in 1981 knew that their newsgroup postings would be indexed, easily searchable and freely available for all time at Google Groups. The principles of data protection [7] are not enough, as information systems tend to be *leaky*, e.g., the California company ChoicePoint actually sold 145 000 people's personal information to Nigerian scammers [24]. Even though compensation may be available through courts, that is a very dramatic and unwieldy option for individuals.

Design of technical systems cannot rely on utility only. Requirements from privacy management should instead be seen the same way legal requirements are seen: as absolutes, not as something that can be subjected to risk/benefit analysis, scoped out or implemented haphazardly. But there is no clear minimum privacy management standard, so real development will make tradeoffs due to finite resources. We argue that such tradeoffs must take into account the non-utility-based values of privacy management.

8 Conclusions and future work

The ideas presented have been useful in our work, are grounded in social psychology and many of them used by others working in ubiquitous computing. Actual interaction design based on them is wraught with tradeoffs between granularity of control and ease of use, accountability and interaction path-length, plausible deniability and robustness. The proposed user interface is one possible way to make these tradeoffs. We will conduct field evaluations of this proposal.

The requirements for presence protocols derived from the analysis give clear guidance for changing the existing XMPP and SIMPLE protocols. The protocols should add support for giving different data to different subscribers. It should be possible for observations to be accountable to the observed, by letting clients fetch presence information from the server, and the server communicating the fetches to the observed. In designing such protocols privacy has to be taken as an unforgiving requirement, not just as another contribution to utility.

There is no ready answer to how well the theory of self-disclosure is actually applicable to automated awareness services. The fact that the disclosure through the service is not the result of any overt action on the part of the discloser makes the process very different from the forms of self-disclosure traditionally studied. We plan to study how presence is used in existing instant messaging services like IRC and whether the phenomena can be explained as self-disclosure. A related open question is *how* reciprocity should be supported: as a mandatory feature of the service or by providing tools for users to make the service reciprocal.

References

1. Raento, M., Oulasvirta, A., Petit, R., Toivonen, H.: Contextphone, a prototyping platform for context-aware mobile applications. IEEE Pervasive Computing **4** (2005) to appear.
2. Dourish, P., Bellotti, V.: Awareness and coordination in shared workspaces. In: Proceedings of the ACM conference on Computer-supported cooperative work, New York, ACM Press (1992) 107–114
3. Langheinrich, M.: Privacy by design – principles of privacy-aware ubiquitous systems. In Abowd, G.D., Brumitt, B., Shafer, S.A., eds.: Proceedings of the Third International Conference on Ubiquitous Computing (UbiComp 2001). Number 2201 in LNCS, Atlanta, USA, Springer-Verlag (2001) 273–291
4. P. Saint-Andre (ed.): IETF. RFC 3920. Extensible messaging and presence protocol (XMPP): Core (2004)
5. IETF Secretariat: SIP for Instant Messaging and Presence Leveraging Extensions (simple). online `http://www.ietf.org/html.charters/simple-charter.html`, referenced 2005-05-05 (2005)
6. Altman, I., Vinsel, A., Brown, B.: Dialectic conceptions in social psychology : an application to social penetration and privacy regulation. Advances in Experimental Social Psychology **14** (1981) 108–161

7. The European Commission: Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data. Official Journal of the European Communities (1995) 31–50
8. Consolvo, S., Smith, I.E., Matthews, T., LaMarca, A., Tabert, J., Powledge, P.: Location disclosure to social relations: Why, when, & what people want to share. In: Proceedings of CHI 2005 Conference on Human Factors in Computing Systems. (2005) to appear.
9. Cozby, P.: Self-disclosure, reciprocity and liking. Sociometry **35** (1972) 151–160
10. Lederer, S., Hong, J.I., Dey, A.K., Landay, J.A.: Personal privacy through understanding and action: five pitfalls for designers. Personal and Ubiquitous Computing **8** (2004) 440–454
11. Andersen, S., Reznik, I., Glassman, N.: The unconscious relational self. In Hassin, R., Bargha, J., Uleman, J., eds.: The new unconscious, New York, Oxford University Press (2004)
12. Oulasvirta, A., Tamminen, S., Roto, V., Kuorelahti, J.: Interaction in 4-second bursts: The fragmented nature of attention in mobile HCI. In: Proceedings of the 2005 Conference on Human Factors in Computing Systems (CHI 2005), New York, USA, ACM Press (2005) To appear.
13. Palen, L.: Social, individual and technological issues for groupware calendar systems. In: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press (1999) 17–24
14. Aoki, P.M., Woodruff, A.: Making space for stories: ambiguity in the design of personal communication systems. In: CHI '05: Proceeding of the SIGCHI conference on Human factors in computing systems, New York, NY, USA, ACM Press (2005) 181–190
15. Aoki, K., Downes, E.J.: An analysis of young people's use of and attitudes toward cell phones. Telemat. Inf. **20** (2003) 349–364
16. Rubin, Z.: Disclosing oneself to a stranger: reciprocity and its limits. Journal of Experimental Social Psychology **11** (1975) 233–260
17. Berg, J.H., Derlega, V.J.: Themes in study of self-disclosure. In Derlega, V.J., Berg, J.H., eds.: Self-Disclosure. Theory, Research and Therapy, New York, Plenum Press (1987) 1–8
18. Prinz, W.: NESSIE: An awareness environment for cooperative settings. In: Proceedings of the Sixth European Conference on Computer Supported Cooperative Work — ECSCW99, Kluwer Academic Publishers (1999) 391–410
19. Isaacs, E., Walendowski, A., Ranganthan, D.: Hubbub: a sound-enhanced mobile instant messenger that supports awareness and opportunistic interactions. In: CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press (2002) 179–186
20. FTC Advisory Committee on Online Access and Security: Final report (2000). `http://www.ftc.gov/acoas/papers/finalreport.htm`
21. Rubinstein, A.: A theorist's view of experiments. European Economic Review **45** (2001) 615–628
22. Loewenstein, G., O'Donoghue, T., Rabin, M.: Projection bias in predicting future utility. The Quarterly Journal of Economics **118** (2003)
23. Acquisti, A., Grossklags, J.: Losses, Gains, and Hyperbolic Discounting: An Experimental Approach to Personal Information Security Attitudes and Behavior. In Camp, J., Lewis, R., eds.: The Economics of Information Security, Kluwer (2004)
24. Perez, E.: Identity theft puts pressure on data sellers (2005). `http://www.post-gazette.com/pg/05052/460233.stm`