A Privacy-Preserving UBICOMP Architecture

Yee, George

Publisher's version / Version de l'éditeur:

Proceedings of the 2006 International Conference on Privacy, Security, and Trust (PST 2006), 2006

NRC Publications Archive Record / Notice des Archives des publications du CNRC :
https://nrc-publications.canada.ca/eng/view/object/?id=b9dcd3ee-caab-4cef-8372-f484091ac81e
https://publications-cnrc.canada.ca/fra/voir/objet/?id=b9dcd3ee-caab-4cef-8372-f484091ac81e

National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de technologie de l'information

NRC · CNRC

A Privacy-Preserving UBICOMP Architecture *

Yee, G.
November 2006

* published in the Proceedings of the 2006 International Conference on Privacy, Security, and Trust (PST 2006). Markham, Ontario, Canada. October 30 – November 1, 2006. NRC 48747.

Canada

A Privacy-Preserving UBICOMP Architecture[1]

George Yee
Institute for Information Technology
National Research Council Canada
george.yee@nrc.ca

Abstract

With the increasing deployment of sensors, intelligent devices of all sizes, and wireless networking, ubiquitous computing (UBICOMP) environments are getting closer to reality. Research in UBICOMP has focused on enabling technologies, such as networking, data management, security, and user interfaces. However, privacy for UBICOMP has been a contentious issue and the privacy concerns that have been raised suggest that privacy may be the greatest barrier to the long-term success of UBICOMP. This paper proposes a hybrid (locally centralized but peer-to-peer across the Internet) UBICOMP architecture that respects personal privacy preferences expressed in the form of personal privacy policies.

Keywords: privacy protection, privacy policy, ubiquitous computing, UBICOMP, architecture

1. Introduction

The popular concept of ubiquitous computing (UBICOMP) began with Mark Weiser's seminal paper [3] in 1991, where he introduced the vision of a person interacting with hundreds of nearby computers wirelessly networked and distributed in his physical environment. Weiser's goal was to emphasize the person rather than the machine, focusing on helping the person in his/her daily life. Weiser's vision has not been realized, but researchers are getting closer. Research has focused on enabling technologies, such as networking, data management, security, and user interfaces [1]. However, privacy for UBICOMP has been a contentious issue and the privacy concerns that have been raised suggest that privacy may be the greatest barrier to the long-term success of UBICOMP [2].

The main objective of this paper is to present an architecture for UBICOMP that preserves personal privacy preferences expressed in the form of personal privacy policies. A secondary objective is to discuss the nature of privacy and how personal privacy preferences may be specified in a privacy policy suitable for UBICOMP.

This work addresses an UBICOMP environment (see Figure 1) with the following characteristics:

- Ubiquitous computing devices (e.g. laptops, PDAs, cell phones, workstations) are locally networked (e.g. Ethernet, Wi-Fi, IrDA, Bluetooth) as well as globally networked via the Internet.
- The local computing devices are owned by a human or an organization.
- Human users employ these devices to share information locally and globally. A user who shares (or sends) information is called a *data sharer*. One who observes (or receives) information is called a *data observer*. A user may be both a data sharer and a data observer.

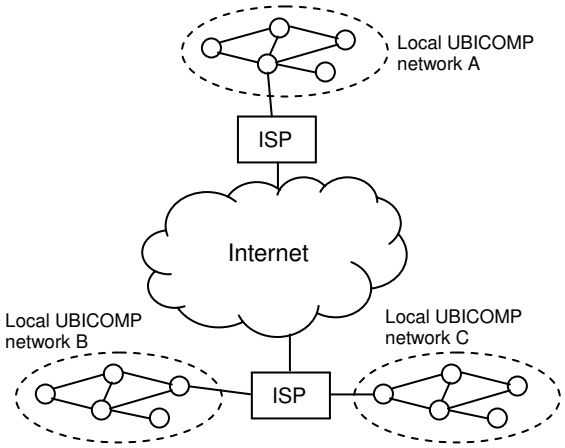

Figure 1. UBICOMP environment (ISP = Internet Service Provider, small circles are UBICOMP devices)

The remainder of this paper is organized as follows. Section 2 examines privacy and the specification of a personal privacy policy suitable for UBICOMP. Section 3 presents the proposed peer-to-peer privacy preserving UBICOMP architecture. Section 4 evaluates the proposed architecture by discussing some strengths and weaknesses. Section 5 examines related work. Section 6 concludes the paper and lists some ideas for future research.

2. Privacy policies for UBICOMP

2.1. Privacy

As defined by Goldberg et al. in 1997 [17], privacy refers to the ability of individuals to *control* the collection, retention, and distribution of information about themselves. This is the definition of privacy used for this work. Protecting an individual's privacy then involves endowing the individual with the ability to control the collection, retention, and distribution of her ("her" and "she" are used here to stand for both sexes) personal information.

2.2. Use of privacy policies

In this work, giving an individual or data sharer control over her private information is achieved as follows. The data sharer specifies in a personal privacy policy how she wants her personal information handled by the data observer; the data observer, on the other hand, specifies in her privacy policy what personal information she requires from the data sharer and how she plans to handle the data sharer's information. The data sharer's policy has to be compatible or match the data observer's policy before information sharing can begin. The matching of privacy policies is outside the scope of this work but see [15]. If the policies do not match, the data sharer can either negotiate with the data observer to try to resolve the disagreement or choose a different data observer. The negotiation of privacy policies is outside the scope of this work but see [11]. Once the sharing begins, the data observer has to comply with her privacy policy (which is compatible with the data sharer's privacy policy). Foolproof mechanisms must be in place to ensure compliance. Such mechanisms are outside the scope of this work. An example of a privacy policy compliance mechanism for e-services is given in [12]. The question of what to specify in a personal privacy policy is taken up next.

2.3. Privacy policies from legislation

Legislative bodies throughout the world [6, 7, 8] have enacted legislation to give the individual control over her personal information as described above. They have defined personal information and spelled out the obligations of a service provider organization with respect to the personal privacy of a service consumer. Such legislation can be used as a basis for defining the content of personal privacy policies. Yee & Korba [11] have defined a personal privacy policy based on Canadian privacy legislation [6, 7], which is representative of principles behind privacy legislation in many countries. This privacy policy (Figure 2) is for users of services from service providers (e.g. e-learning service provider).

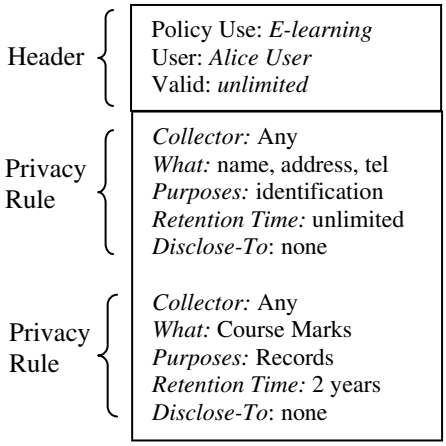

Figure 2. Example personal privacy policy based on Canadian privacy legislation [6, 7].

As shown in Figure 2, a personal privacy policy for services consists of a header section followed by one or more privacy rules, where there is one rule for each item of private information. The header fields have the following meaning: *Policy Use* identifies the service application, *User* gives the name of the user who owns the policy, and *Valid* indicates the period of time during which the policy is valid. The fields in each privacy rule have the following meaning: *collector* identifies the service provider that wishes to collect the information, *what* describes the nature of the information, *purposes* identifies the purposes for which the information is being collected, *retention time* pinpoints the amount of time for the service provider to keep the information, and *disclose-to* identifies any other parties who will receive the information.

The content of the above personal privacy policy is generic since it is based on legislation that applies across the board. It can be specialized to UBICOMP by testing it against a set of questions (Table 1) given by Hong et al. [2] for privacy risk analysis of UBICOMP. This testing will identify the extent to which the policy contents address the privacy risks of UBICOMP as well as any additions needed to address any remaining risks. The result will be a personal privacy policy for UBICOMP that satisfies legislative requirements. Hong et al organized their questions into two groups: one group looking at the social and organizational context in which an application is embedded, the other group examining the technology used to implement the application. "PRAQ" is the name used to refer to these questions; PRAQ.*n* refers to question *n* within PRAQ.

Table 1. PRAQ – Privacy risk analysis questions for UBICOMP from [2]

	Social and Organizational Context
1.	Who are the users of the system? Who are the *data sharers*, the people sharing personal information? Who are the *data observers*, the people that see that information?
2.	What kinds of personal information are shared? Under what circumstances?
3.	What is the value proposition for sharing personal information?
4.	What are the relationships between data sharers and data observers? What are the relevant level, nature, and symmetry of trust? What incentives do data observers have to protect data sharers' personal information (or not, as the case may be)?
5.	Is there the potential for malicious data observers (e.g., spammers and stalkers)? What kinds of personal information are they interested in?
6.	Are there other stakeholders or third parties that might be directly or indirectly impacted by the system?
	Technology
7.	How is personal information collected? Who has control over the computers and sensors used to collect information?
8.	How is personal information shared? Is it opt-in or is it opt-out (or do data sharers even have a choice at all)? Do data sharers push personal information to data observers? Or do data observers pull personal information from data sharers?
9.	How much information is shared? Is it discrete and one-time? Is it continuous?
10.	What is the quality of the information shared? With respect to space, is the data at the room, building, street, or neighborhood level? With respect to time, is it real-time, or is it several hours or even days old? With respect to identity, is it a specific person, a pseudonym, or anonymous?
11.	How long is personal data retained? Where is it stored? Who has access to it?

We consider each question in turn, as follows:

- PRAQ.1: The users of the system are the data sharers and the data observers. However, it is necessary for privacy (and required by privacy legislation) that the data sharers know the identities of the data observers and this is not reflected in our collector attribute, which refers to the service provider. Thus we make the following change: replace "Collector" with "Data Observer".
- PRAQ.2: This is taken care of by our policy using the "what" and "purposes" attributes.
- PRAQ.3: The value proposition is reflected in the "Policy Use" attribute. Users would engage the ubiquitous system only if they receive some value in doing so.
- PRAQ.4: This question assesses the level of trust between data sharers and data observers to see if data sharers would be comfortable in sharing their information. In addition, the question attempts to reinforce that trust by asking if the data observer has incentives to protect the data sharers' information. In our use of privacy policies, the data observer has to comply with the data sharer's privacy policy and foolproof compliance mechanisms are required to be in place. Therefore, data sharers are assured that their wishes are respected. Trust is still relevant for us, since it will partially determine whether or not the system will be used. Trust will be reflected in the choice of data observers specified by data sharers in their privacy policies.
- PRAQ.5: This question is intended to determine if the private information shared needs protection. This question is taken care of by privacy legislation which requires a) that private information be protected using appropriate security mechanisms, and b) that data observers must comply with the data sharer's privacy policy (ensuring compliance is another matter that requires foolproof compliance mechanisms).
- PRAQ.6: This question aims to find out if other stakeholders or third parties could suffer some loss of privacy due to the way the system works. This is a valid question that should be answered and appropriate remedies taken prior to system deployment. However, it does not require a change to the privacy policy.
- PRAQ.7: The data sharer supplies the personal information when requested by the system. The owner of the ubiquitous system controls the computers and sensors used to collect personal information. This question does not impact the privacy policy.
- PRAQ.8: A data sharer can opt-out (try a different data observer) if there is no match between her privacy policy and the data observer's privacy policy. Data observers pull information from data sharers. This question does not impact the privacy policy.
- PRAQ.9: The information shared can be discrete, one-time, or continuous. This question also does not require changes to the privacy policy.
- PRAQ.10: This question assesses the quality of the information to see if it needs protection in terms of risk. It is like PRAQ.5. All shared private information is protected according to the sharer's privacy policy. This question does not require changes to the privacy policy.
- PRAQ.11: Retention time is specified in the privacy policy. Data observers specified in sharers' policies together with the sharers have access. The data is

stored in either a central, distributed, or combined central and distributed fashion, as determined by the design of the ubiquitous system.

The above tests of the services privacy policy have revealed that only a minor change is needed due to PRAQ.1. Also, the user field in the header is changed to "Data Sharer". Figure 3 shows the format of the data sharer personal privacy policy for UBICOMP along with a matching (because the retention times required by the observer are compatible with the retention times offered by the sharer) data observer privacy policy.

3. Privacy-preserving UBICOMP architecture

3.1. Architecture requirements

Given the UBICOMP environment and the use of privacy policies to protect a data sharer's privacy, the requirements of the architecture are as follows:

1. The UBICOMP environment is as describe in section 1. The devices within this environment are assumed to be capable of performing the tasks ascribed to devices below.
2. The use of privacy policies is as described in section 2.2 (including privacy policy compliance).
3. Prior to using a UBICOMP device to share or observe data, the user logs into the device and identifies herself.
4. Information regarding which data observers are online can be called up at any time on any UBICOMP device.
5. After logging in and prior to sharing data, the data sharer performs the following steps:
    a) Requests to see which data observers are online.
    b) Completes her privacy policy (previously stored as a template without any data observers identified) by deciding which data observers will receive her private information based on policy use. Submits her privacy policy to the UBICOMP system.

    The system will automatically request the required data observer policies and check for policy compatibility or matching; for each pair of policies compared, the system optionally allows the sharer to negotiate with the observer if the policies do not match. For each match, the system automatically sets up a connection to the data observer with the matching policy for data sharing. Once the data sharing session is finished, the system automatically tears down the associated connections.
6. After logging in and prior to observing data, the data observer submits her privacy policy to the system when requested to do so.
7. A user who is both a data sharer and a data observer does both of the previous two items for data sharers and data observers.

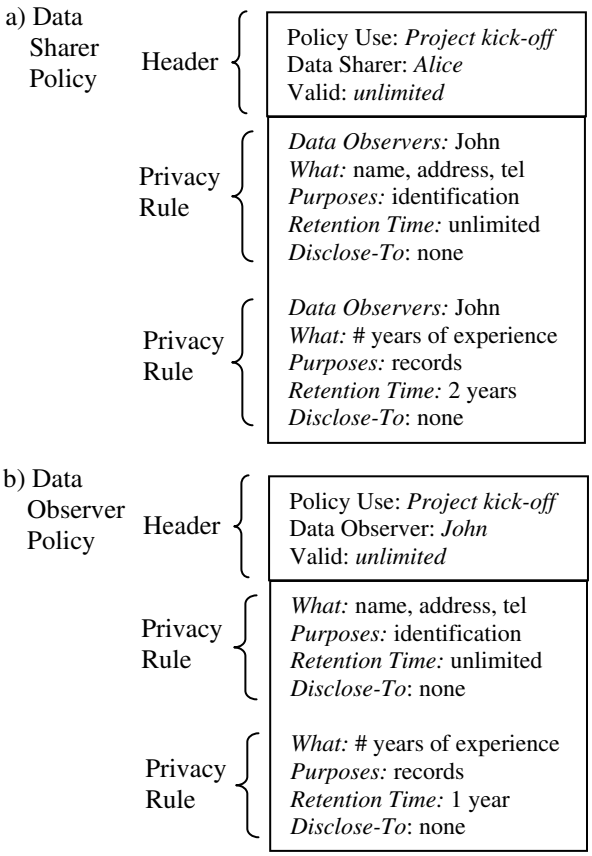

Figure 3. Example data sharer personal privacy policy and matching data observer privacy policy for UBICOMP

3.2. Architecture design

This section describes the design of the privacy-preserving UBICOMP architecture to satisfy the above requirements. The design has the following components: a) local and global networking as shown in Figure 1, b) a *Privacy Controller* for each local network, and c) interfaces to connect between each local device and the Privacy Controller. The description of each component follows.

Local and global networking: These are assumed to be what is most commonly available, i.e. Ethernet, Wi-Fi, IrDA, or Bluetooth for local, and the Internet for global.

The connection of local networks to the Internet is also assumed to be standard.

Privacy Controller: This component has the following functions:

a) Requests privacy policies from data sharers and data observers specified in the sharer policies. Receives these privacy polices and determines if they match; optionally sets up a privacy policy negotiation between a data sharer and a data observer for a particular policy pair that does not match. These functions are carried out by a *Policy Module (PM)* within the Privacy Controller.

b) Ensures that a data observer complies with the privacy policy of a data sharer. This function is done by a *Compliance Module (CM)* within the Privacy Controller (similar to [12]).

c) Sets up connections for data sharing between data sharers and data observers who have matching privacy policies. Tears down the connections once the associated sharing sessions are finished. This function is performed by a *Link Module (LM)* within the Privacy Controller.

d) The Privacy Controller of each local network performs the above for data sharer-data observer pairs within the Controller's local network. For a data sharer who is sharing data with a data observer in a different local network, the Privacy Controller of the data sharer's local network and the Privacy Controller of the data observer's local network must communicate with each other to perform the above (e.g. *PM* of the remote data observer's local network passes the data observer's privacy policy to the *PM* of the data sharer's local network).

Interfaces: Devices in the local environment other than the Privacy Controller need to have appropriate interfaces that inter-work with the Privacy Controller to carry out privacy policy management (e.g. privacy policy submission, connection setup for sharing, policy compliance checking). For devices with very limited computational capability (e.g. embedded or wearable), these interfaces will have to be commensurate with the computational capability of each device (for these devices the quantity of shared private information will be limited too).

Figure 4 illustrates the proposed privacy-preserving UBICOMP architecture. It can be easily seen that this is a hybrid architecture that is globally peer-to-peer, with peer nodes being the local networks, but within each local network, the UBICOMP devices are centralized to the Privacy Controller.

The behaviors of the PM and LM components of the Privacy Controller are illustrated by the high level state machines in Figure 5 (the CM is outside the scope of this work). The arrows in Figure 5 are labeled using the convention "condition / action". Further, the symbol "?" stands for "received" and "!" stands for "send". In Figure 5(a), the PM detects that a new data sharer has logged in and requests the sharer for her privacy policy. Once this policy is received, the PM requests the privacy policies of all data observers specified in the sharer's privacy policy. The PM then compares each received observer policy with the sharer's policy to determine if there is a match. If there is a match, the match is registered and information about the matching sharer and observer is sent to the LM for connection establishment. If there is no

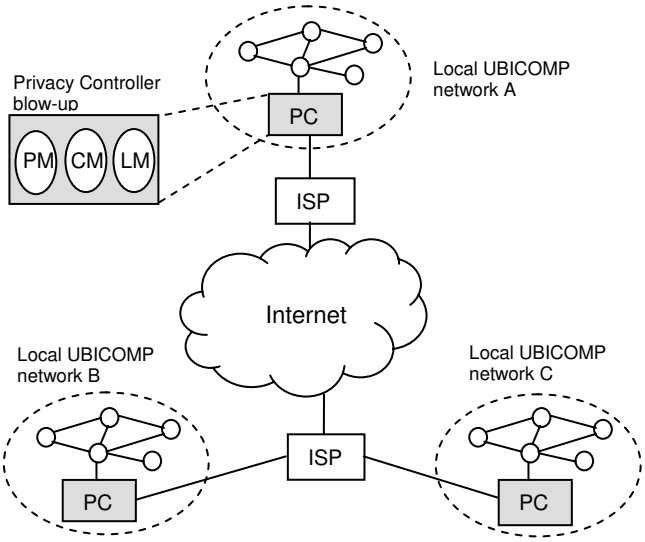

Figure 4. Privacy-Preserving UBICOMP architecture (ISP = Internet Service Provider, small circles are UBICOMP devices, PC = Privacy Controller, PM = Policy Module, CM = Compliance Module, LM = Link Module).

match, the sharer has the choice to negotiate with the observer to resolve the mismatch or allow the mismatch to stand (in which case the PM proceeds to process the next observer's policy). If negotiation is chosen and succeeds, the resulting match is registered and the LM is notified as previously mentioned. If negotiation is chosen and fails, the mismatch has to stand and the PM proceeds to process the next observer's policy. In Figure 5 (b), upon receiving policy match information from the PM, the LM proceeds to use the match information to set up a connection between the data sharer and the matching data observer. Once the connection is established, the LM notifies the sharer and the observer that they are connected and the sharer's session with observers can begin (first observer connected) or continue (other

observers already connected). Upon receiving a signal from the sharer's device that the session has finished, the LM tears down all associated connections.

Figure 6 presents a message sequence chart showing the interactions between a data sharer, the PM, the LM, and a data observer (only one is shown for simplicity) for the period between the point when a data sharer logs in to the point when she has completed her data sharing session. The scenario depicted is for a first time successful policies match.

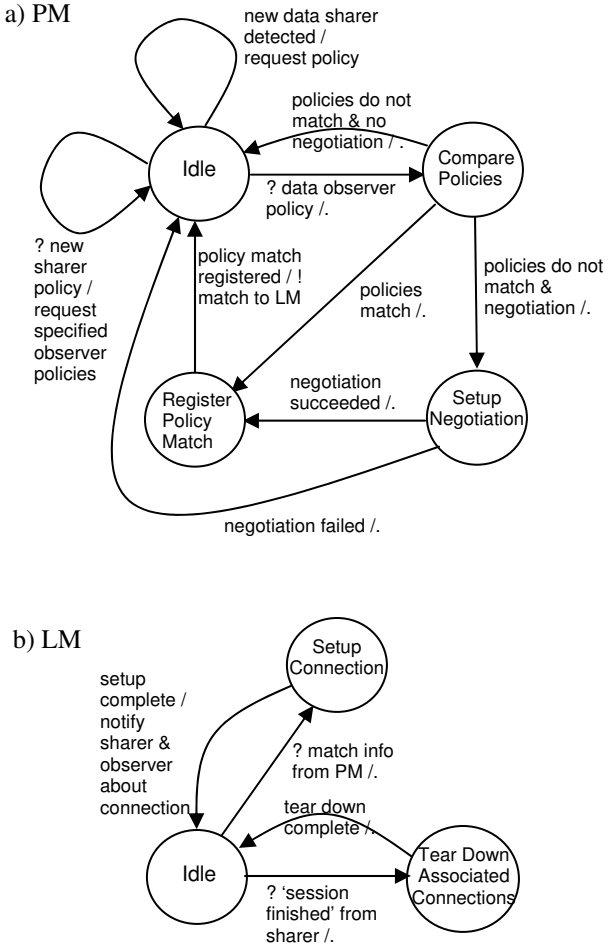

Figure 5. High level state machines illustrating the behavior of the PM and LM components of the Privacy Controller.

Table 2 shows which architecture entities are responsible for satisfying the requirements in section 3.1.

3.3. Implementation notes

Some implementation aspects of the architecture are considered here.

How does the data sharer come up with her personal privacy policy? It is proposed that data observers routinely advertise their data requirements on the Internet. Note that this is in a way being done today by service websites (e.g. when the user is asked to fill out an online form). Data sharers can then formulate and store a policy template that specifies their privacy preferences given the data requirements of the observers. For their policies to be complete, data sharers need only add at runtime the observers with whom they are willing to share their data. Alternatively, the sharers can employ a scheme similar to [9], where the privacy rules can be selected according to the level of privacy desired using a privacy slider.

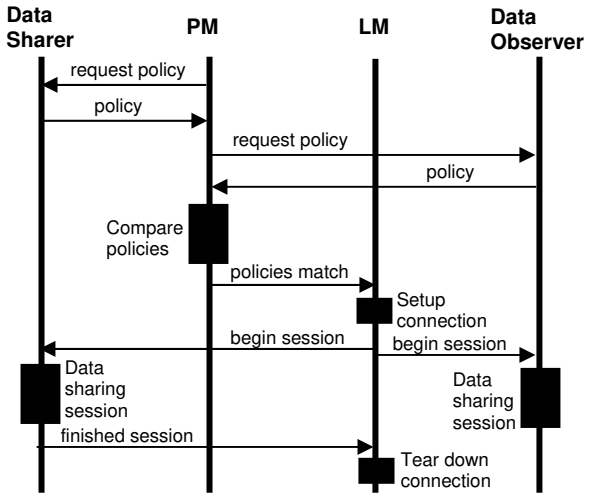

Figure 6. Message sequence chart showing the interactions leading up to a successful policies match and the ensuing session.

Table 2. Responsibility for satisfying architecture requirements

Architecture Requirement	Entity or Module that Satisfies the Requirement
1	Entire Architecture
2	Entire Architecture
3	Device and Network Software
4	Device and Network Software
5	Device and Network Software, PM and LM modules of the Privacy Controller
6	Device and Network Software
7	Device and Network Software, PM and LM modules of the Privacy Controller

The heterogeneous nature of UBICOMP devices may present some implementation problems for the proposed

architecture due to insufficient computing power to host required software for the device, the network, and the interfaces. Devices that are problematic in this respect will have to be excluded from the architecture. It is assumed that this potential loss is acceptable because such devices would probably not be capable of communicating very much information anyway.

Due to its intensive computational requirements (e.g. comparing privacy policies, multitasking between numerous sharer-observer pairs), the Privacy Controller should be hosted on a workstation of sufficient power that's part of the local UBICOMP environment.

The display of data observers online (requirement 4 of section 3.1) may include a reputation value for each data observer. This would help the data sharer choose which data observer to include in her privacy policy. The reputation value may be calculated based on the observer's history of past transactions, as is done on eBay.com.

What does matching of policies mean? There needs to be a way of comparing two policies using some measure of compatibility such as levels of privacy [15]. However, the UBICOMP policies above need to be compared based on the "*Data Observers*" field as well, which is an added dimension not covered in [15]. However this added requirement can be easily satisfied by a literal or fuzzy comparison.

Protocols need to be defined for the policy management messages between the Privacy Controller and the UBICOMP devices.

Privacy policies need to be amenable to machine processing. Policy languages such as APPEL [14] and EPAL [16] that are XML-based are good choices.

Finally, the data sharer's private information and her privacy policies need to be secured from attack (required by privacy legislation). In addition, users will need to be authenticated. Appropriate security mechanisms will need to be applied or developed and applied, such as the use of data encryption to encrypt the private information, and the use of certificates for authentication. In addition, the Privacy Controller and in particular, the CM, need to be protected from malicious tampering. Since the CM plays the all important role of checking for compliance, it may be implemented in hardware to resist tampering. The entire Privacy Controller may be certified by a trusted authority to increase user trust.

3.4. Application example

Suppose Alice wishes to have a consultation with three online doctors about her medical condition. She wishes to consult with all of the doctors at the same time so that she can compare responses in real time and be able to ask new questions based on this comparison. She is in an office with a UBICOMP environment that has several UBICOMP devices, including a WI-FI connected laptop. The devices are locally networked and form part of a privacy-preserving architecture. Alice begins her simultaneous sessions with the three doctors after the following steps:

1. Alice logins to the UBICOMP system (note: "system" and "environment" are used synonymously here) using the laptop and identifies herself. After she is authenticated by the system, she requests to see all doctors who are online. The system responds with a listing of online doctors along with their specialties and reputations.
2. Alice retrieves her pre-specified privacy policy from the system and completes it by choosing and including three online doctors, based on their reputations and medical specialties. Alice submits her privacy policy to the system.
3. The system requests the privacy policies of the online doctors that Alice specified in her privacy policy. With the arrival of each doctor's policy, the system compares Alice policy with the doctor's policy to see if the policies match-up. All doctors' policies match except for one. Alice is asked if she wants to negotiate with the non-matching doctor to try to resolve the non-match. Alice agrees to negotiate and is able to negotiate to a successful conclusion. Now all policies match.
4. The system sets up connections between Alice and the three online doctors. Alice and the doctors are told that they may begin their sessions.

During the sessions, Alice is asked to share personal information with the doctors, such as her date of birth, her history of personal illnesses, and the medical condition for which she seeks advice. In addition, during and after the sessions, the CM modules of the doctors' respective Privacy Controllers, continuously checks the doctors' handling of Alice's personal information to ensure compliance with Alice's privacy policy. The CM modules log all data activities to a secure log (as done in [12]) which can be verified for compliancy should the doctors' data handling ever be challenged.

4. Evaluation

Some strengths of the proposed UBICOMP architecture are: a) upholds personally specified privacy preferences, b) can be used for all types of session computing, and c) high scalability. Some weaknesses are: a) may be difficult to retrofit the proposed architecture into an already existing architecture, and b) since the proposed architecture encompasses the entire Internet, it may be difficult for users interacting from countries with

very different cultures to come to agreement over privacy policies. These points are elaborated below.

In terms of the strengths, the proposed architecture allows each user to specify her privacy preferences in a privacy policy and for this policy to be upheld. Further, disagreements in privacy policies may be negotiated. Next, the architecture allows a privacy-preserving "session" to be set-up between a data sharer and a data observer. It leaves open what computing can be done in the session. Therefore, the session can be an e-commerce session where the data sharer is a buyer and the data observer is a seller, for example, or any other type of session that requires privacy protection. Finally, the architecture is highly scalable. Each privacy controller serves only its local network, or communicates with other privacy controllers in case the data sharer and data observer are from different local networks. If a local network gets too large, it can simply be divided into two or more local networks, each network with its own privacy controller. In the case where a single data sharer could have many data observers, the data sharer is in control of how many data observers she wants to deal with – she will only deal with as many as she can humanly handle, which can be used to configure the number of devices in a local network to ensure adequate performance.

In terms of the weaknesses, it is true that it may be difficult to retrofit an already existing architecture into the proposed architecture. On the other hand, existing UBICOMP architectures that can be classified as ad hoc networks should be easier to retrofit. The proposed architecture is best applied to new UBICOMP environments. Next, the difficulty of obtaining privacy policy agreement between users of culturally distinct countries is not really a problem that is unique to the proposed architecture. It would be a problem of any architecture that brings such users together. Rather, the proposed architecture benefits such users by allowing them to negotiate in the first place, which should be better than not having the capacity for negotiation.

5. Related work

In the privacy literature for UBICOMP, Hong and Landay [4] focus on providing tools for designing individual privacy-sensitive applications, while Hong et al. [2] suggest the use of privacy risk models to make application designers aware of the privacy concerns and risks in their design. Di Pietro and Mancini [5] have considered broad measures for upholding privacy, such as the use of logical borders to limit propagation of information and the application of anonymous user identities to protect the real users. This work is a departure from the above works in the sense that privacy protection is driven by privacy policies across all applications of data sharing, rather than focusing on the design of individual applications to respect privacy.

Yee [18] presented a privacy policy approach for preserving privacy in UBICOMP that shares some similarities with this work. However, the UBICOMP environment in [18] is only comprised of an organization's closed network of UBICOMP devices with no external networking. Moreover, the comparison of privacy policies in [18] is between the privacy policy of a data sharer/observer and the privacy policy of the organization for a particular device (the organization owns the devices) that the data sharer/observer is interested in using. A data sharer's/observer's privacy policy must be compatible with the organization's privacy policy for the device before that device can be used. Thus, the UBICOMP and privacy models in [18] are considerably different from their counterparts in this work.

Other related recent works include Ackerman [19], who proposes labeling protocols that can be used to notify the user of potential data capture, and Dragovic & Crowcroft [20], who propose mitigation of privacy risks through data manipulation. Finally, there are works targeted at Internet e-services environments dealing with privacy policy derivation [9], privacy policy negotiation [10, 11], privacy policy compliance [12], and treating the protection of privacy as a kind of rights management [13].

Di Pietro and Mancini [5] make an interesting comment (not pursued in their paper) that portends the approach of this work. They write in their conclusion: "Finally, we emphasize the need for an easy to configure and manageable personal profile to control the interactions among the many HWW devices that could surround a user. The enforcement of such a profile could be a means to preserve the user's personal privacy." "HWW" stands for hand-held/wearable wireless devices. Their "personal profile" sounds very much like a personal privacy policy.

6. Conclusions and future research

This work has proposed a hybrid, globally peer-to-peer, locally centralized privacy-preserving architecture for UBICOMP. In this architecture, privacy is protected through compliance with privacy preferences expressed as personal privacy policies. This paper has also defined the content of such policies for use in the architecture. The use of privacy policies, together with suitable compliance mechanisms, appears to be an effective way to protect privacy in a ubiquitous computing environment. Such use gives the data sharer flexibility and control over her private information, and inspires her to trust the system.

The applications for the architecture proposed in this work are many, including e-commerce (data observers are service providers, data sharers are service consumers) and e-collaboration (every user is both a data sharer and a data observer).

Future work includes the construction of a prototype to fine-tune the above design and investigate some of the implementation details discussed in the implementation notes. More research is needed into how privacy controllers can cooperate to serve data sharers and observers that interact from different local networks. An application work item could be the design of an e-collaboration system applying the architecture proposed here.

Acknowledgements

The author gratefully acknowledges the support of the National Research Council Canada for this work.

References

[1] P. Boddupalli, F. Al-Bin-Ali, N. Davis, A. Friday, O. Storz, and M. Wu, "Payment Support in Ubiquitous Computing Environments", Proceedings of the Fifth IEEE Workshop on Mobile Computing Systems & Applications (WMCSA 2003), Monterey, California, Oct. 9-10, 2003.

[2] J.I. Hong, J.D. Ng, S. Lederer, and J.A. Landay, "Privacy Risk Models for Designing Privacy-Sensitive Ubiquitous Computing Systems", Proceedings, ACM conference on Designing Interactive Systems (DIS2004), Cambridge, Massachusetts, August 1-4, 2004.

[3] M. Weiser, "The Computer for the Twenty-First Century", *Scientific American*, September 1991, pp. 94-100.

[4] J.I. Hong and J.A. Landay, "An Architecture for Privacy-Sensitive Ubiquitous Computing", Proceedings, The Second International Conference on Mobile Systems, Applications, and Services (MobiSys2004), Boston, Massachusetts, June 6-9, 2004.

[5] R. Di Pietro and L.V. Mancini, "Security and Privacy Issues of Handheld and Wearable Wireless Devices", *Communications of the ACM*, Vol. 46, No. 9, Sept. 2003.

[6] Department of Justice, "Privacy Provisions Highlights", http://canada.justice.gc.ca/en/news/nr/1998/attback2.html (visited Sept. 12, 2003).

[7] Canadian Standards Association, "Model Code for the Protection of Personal Information", http://www.csa.ca/standards/privacy/code/Default.asp?articleID=5286&language=English (visited Sept. 5, 2003).

[8] European Union, "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data", unofficial text, http://aspe.hhs.gov/datacncl/eudirect.htm (visited Sept. 5, 2003).

[9] G. Yee, L. Korba, "Semi-Automatic Derivation and Use of Personal Privacy Policies in E-Business", *International Journal of E-Business Research*, Vol. 1, No. 1, pp. 54-69, Idea Group Publishing, 2005.

[10] G. Yee, L. Korba, "Bilateral E-services Negotiation Under Uncertainty", Proceedings, The 2003 International Symposium on Applications and the Internet (SAINT2003), Orlando, Florida, Jan. 27-31, 2003.

[11] G. Yee, L. Korba, "The Negotiation of Privacy Policies in Distance Education", Proceedings, 14th IRMA International Conference, Philadelphia, Pennsylvania, May 18-21, 2003.

[12] G. Yee, L. Korba, "Privacy Policy Compliance for Web Services", Proceedings, 2004 IEEE International Conference on Web Services (ICWS 2004), San Diego, California, July 6-9, 2004.

[13] S. Kenny and L. Korba, "Adapting Digital Rights Management to Privacy Rights Management", Computers & Security, Vol. 21, No. 7, November 2002, 648-664.

[14] W3C, "A P3P Preference Exchange Language 1.0 (APPEL 1.0)", http://www.w3.org/TR/P3P-preferences/ (visited Apr. 22, 2004).

[15] G. Yee, L. Korba, "The Comparison of Privacy Policies", Proceedings, 16th IRMA International Conference, San Diego, California, May 15-18, 2005.

[16] Enterprise Privacy Architecture Language (EPAL), http://www.zurich.ibm.com/security/enterprise-privacy/epal/ (visited Oct. 10, 2003).

[17] I. Goldberg, D. Wagner, and E. Brewer, "Privacy-Enhancing Technologies for the Internet", *IEEE COMPCON'97,* pp. 103-109, 1997.

[18] G. Yee, "Using Privacy Policies to Protect Privacy in UBICOMP", The First International Workshop on Ubiquitous Smart Worlds (USW2005) held in conjunction with AINA 2005, in Proceedings of AINA 2005, Vol. 2, pp. 633-638, Tamkang University, Taiwan, March 28-30, 2005.

[19] M. Ackerman, "Privacy in Pervasive Environments: Next Generation Labeling Protocols", *Personal Ubiquitous Computing*, 2004 (8): 430-439, Springer-Verlag.

[20] B. Dragovic, J. Crowcroft, "Information Exposure Control through Data Manipulation for Ubiquitous Computing", Proceedings, NSPW 2004, Nova Scotia, Canada.

---

[1] NRC Paper Number: NRC 48747