Using User Preferences to Enhance Privacy in Pervasive Systems

Elizabeth Papadopoulou, Sarah McBurney,
Nick Taylor, M. Howard Williams
Heriot-Watt University, Riccarton,
Edinburgh, UK
{ceeep1 ceesmm,nkt, mhw}@macs.hw.ac.uk

Kajetan Dolinar
SETCCE, Ljubljana,
Slovenia
kajetan@e5.ijs.si

Martin Neubauer
IKR. Univ. Stuttgart
Stuttgart, Germany
martin.neubauer@ikr.uni-stuttgart.de

Abstract

With the increasing interest in developing pervasive computing technologies there is growing recognition of the problems of maintaining user privacy. In the Daidalos pervasive system this is achieved primarily through the use of virtual identities, which are used to conceal the real identity of the user. One problem with this lies in determining to what extent the user should be engaged in the decisions relating to the selection of virtual identities, and what can be done automatically. The solution lies in creating a set of user preferences to assist in taking these decisions, refining them through the use of machine learning techniques. This paper outlines the approach being investigated and describes how this will be achieved when the processes involved in building up user preferences are not trusted.

1. Introduction

In 1991 Weiser [1] presented a vision in which the environment surrounding the user would be filled with computing entities, supporting the user in a variety of ways without continual direction. Since then developments in areas such as sensor technologies, communications, smart dust and motes [2, 3] and specks [4], etc., are enabling these predictions to be realised. Although there is still a long way to go, there is now a growing view that during the next decade or so many of the problems still facing ubiquitous and pervasive computing will be solved and by 2020 this technology will be a reality. However, some major problems still lie ahead and some of the global challenges of the next decade [5] lie in this area.

One of the major problem areas is that of privacy. Both users and services need to know which services they can trust and what information they can share with them. One part of this is concerned with authentication – validating the credentials of a service and ensuring that it is what it claims to be. Another major part is concerned with authorisation – deciding who should be given access to what. In this case it is the identity of the user which is at issue and it is the latter which is the focus of this paper.

Daidalos is a European research project, a major aim of which is to develop a pervasive system [6], focussing especially on mobile users. Security and privacy are key components in this development. This paper is concerned with one of the major aspects of the problem of authorisation in such a pervasive system and the way in which user preferences and learning can be used to support and enhance it.

The next section provides a brief overview of the Daidalos pervasive system. Section 3 discusses privacy and pseudonymity and the notion of virtual identity. Section 4 describes how user preferences and personalization can assist in the automatic selection of virtual identities. Section 5 presents the content and format of the user preferences for virtual identity selection in Daidalos. Section 6 gives an overview of how these user preferences are created and maintained. It also examines the differences and commonalities between the components catering for such preferences and those responsible for ordinary user preferences. Section 7 follows with a summary and conclusion.

2. The Daidalos Pervasive System

Daidalos is a large European research project, whose overall aim is to create a pervasive environment for mobile users. This is achieved by integrating a range of heterogeneous networks and devices and creating a pervasive system on top of this which will protect the user from the complexity of the underlying infrastructure while providing personalized and context aware services with minimal user intervention. The research is divided into two phases with slightly different objectives, spread over a five year period. The work is now in its final stages, with demonstration of the system due in April 2008 and the project as a whole due to finish in December 2008.

The pervasive system (or pervasive service platform) is based around the following six functions:

(1) Service Discovery and Selection. The user can make a request for particular services through a Service Browser. When such a request is issued, the system requires a service discovery functionality to find appropriate services that might be available which could be used to satisfy the user's request. It also needs a Service Selection function to remove services that do not meet the user's preferences (filter them) and order the resulting list according to the user's preferences (rank them).

(2) Service Composition. This functionality is required when the user request requires two or more services to be assembled together to create a composite service that will meet the request.

(3) Session Management. Once the services have been composed, this functionality is required to set them running and to stop them if this becomes necessary. More details on Service Selection and Service Composition are provided in [7].

(4) Personalization. This is a set of functionality concerned with capturing, managing and applying user preferences at various points in the process of providing user services. These include the filtering and ranking of services, personalizing third party services, learning new and managing existing user preferences, etc. Further details on the personalization functions in Daidalos are given in [8].

(5) Context Management. This is responsible for managing the context data relating to the user (e.g. location) as well as to the available services and resources.

(6) Security and Privacy. As previously mentioned user privacy is a priority area in Daidalos, and this functionality [9] affects both knowing who is running what services and controlling access to user data.

The first phase of the project focused on the development of a basic set of functionality to cover the above six functions. As far as ensuring the privacy of the user is concerned, the work done in this phase was limited whereas in the second phase much greater effort is devoted to it. In the second phase it is also assumed that some of the functions might be provided by different service providers. This has a serious consequence if some of the basic system modules are no longer trusted components and the user's identity needs to be concealed from them.

3. Protecting Privacy

Privacy can be regarded as "the right of individuals to protect their ability to selectively reveal information about themselves" [10]. Much work has been done on privacy in the context of the Web and four specific requirements for designing privacy protection have been identified. These are: anonymity, pseudonymity, unlinkability and unobservability [11, 12]. Pervasive systems have a lot in common with the Web and the same requirements apply.

A number of papers (e.g. [13], [14], [15]) have been written on the design of privacy aware ubiquitous systems, reporting on their analysis of end-user requirements and the approaches they follow in order to satisfy them.

One of the important requirements is that there should be simple and appropriate mechanisms for the user to control the release of information. To this end the notions of pseudonymity and anonymity have been adopted.

Pseudonymity is used as a tool to hide the user's identity from services and in so doing conceal the user's digital trail in a pervasive world. At the same time, a pervasive system that allows such mechanisms, should also cater for accountability and should provide mechanisms to protect the user's privacy without encouraging the user to avoid being held accountable for his/her actions [10].

Pseudonymity is useful in online transactions since not every service that is being used needs to identify the user. Authentication does not imply identification. The notion of separate personas, private and public, have been proposed [13] which place different restrictions on the information they release to services. This concept is similar to that of virtual identities, in which the user has a number of virtual identities to protect their real identity. One difference between them is that personas are created based on user preferences and service trust levels while virtual identities are created to match service trust levels and service privacy policies.

Anonymisation goes one step further. Kobsa and Schreck [16] state that anonymisation hides the relationship or linkage between an individual user and his/her stored personal data. With anonymisation, users are never identified and while this works for privacy, it does not allow dynamic personalization or learning of user preferences. Anonymity also creates more problems than it solves due to the fact that it cannot provide accountability [10]. On the other hand, pseudonymity provides a balance between protecting the user's privacy while at the same time offering advanced personalization practices. By using different pseudonyms for different service transactions, pseudonymity provides additional protection to the user's privacy as it partitions the user's interactions and thus hides any direct link between those interactions [17].

Pseudonymity is not sufficient unless unlinkability and unobservability are also satisfied as requirements. If pseudonyms of a user can be linked to each other then the transactions made with one pseudonym belong to the same user that made the transactions with the rest of the linked pseudonyms. This results in gathering of a vast amount of information about the activities of the user, allowing access to the identity of a user from unauthorized services and revealing personal data to unauthorized parties. Unobservability requires that any attacker monitoring the users' interactions cannot identify which interactions belong to the same user [17]. Unobservability becomes more crucial as a requirement when thinking in terms of the user of a pervasive system. The user can be monitored more easily than a user of a traditional system because of the amount of context information maintained about the user in the system.

The next section discusses how user preferences can be employed to enhance pseudonymity and satisfy all of the above privacy requirements in the Daidalos II architecture.

4. Using User Preferences to Select Virtual Identities

Pseudonymity is achieved in Daidalos through the use of multiple identities or Virtual Identities (or VIDs). The initial architecture of the Daidalos Virtual Identity Model is given in [18]. These Virtual Identities form subsets of the user's profile and are used to authenticate the user with services. For any user the set of VIDs may be viewed as a set of different user names, which the user may use for different purposes, and which may conceal all or part of his/her real identity. Each user may have any number of VIDs.

None of the user's Virtual Identities can be linked to any of the others so that if a user uses two virtual identities with the same service, that service will treat these as two different users. This also allows for good personalization practices because users can use services for different activities and have different preferences for each activity. By not providing a direct link between all the services used by a user, user monitoring services will not be able to trace all of the user's transactions, and as a result the user's privacy is protected.

Although the services that the user may use can only see the user's virtual identity and whatever subset of personal information the user allows, deep within the system in the Security and Privacy component the virtual identities can be mapped to real identities for the purposes of accounting.

However, this approach does present a problem for gathering user preferences. If each virtual identity has its own set of user preferences with no connection between them, the task of learning new preferences is made considerably harder and inconvenient to the user.

This problem is overcome in Daidalos by allowing preferences to be shared between virtual identities. To do this without providing any kind of linkage between the virtual identities, the referencing URIs that point to the actual preference location are hidden from the user. In each virtual identity, whenever the user or a service acting on behalf of the user wishes to access the user preferences, the externally accessible URI is replaced with the hidden one. If the user has indicated to the Security subsystem that two or more virtual identities should have a common set of preferences, the hidden URIs for these will point to the same preference location. The mapping between the real referenced URI and the hidden ones is done in the Security and Privacy subsystem which is a trusted subsystem and has access to all Virtual Identities belonging to the user.

When the user switches on the system and authenticates him/herself a default VID is used. Once the user is authenticated, he/she can request a service. In setting up to use the service an appropriate VID needs to be selected for the purpose.

A VID may be created in one of two ways. It may be created explicitly by the user (using a Graphical User Interface) or implicitly by the user setting up specific preferences that allow the system to create a VID based on these preferences and to be used in specific contexts. Selecting a VID to be used presents more challenges than creating it. However, creating a VID can be a part of the process of selecting a VID as will be presented later.

Initially one can make the simple assumption that the user will always select the appropriate VID before requesting any service. However, this can become an arduous task for the user, especially if the number of VIDs grows.

The situation is more complex if one takes account of changing context conditions. Consider the case of a mobile user who is using a network service. As the user moves around he/she may need to change to a different network service (because of falling Quality of Service on the current network due to wireless reception difficulties, increased network traffic, etc., or simply the availability of a more preferable network service). It is not sensible to interrupt the user to ask whether he/she wants to change and, if so, what VID should be used.

Thus in order to provide a user-friendly pervasive environment, the system itself should manage the automatic selection of VIDs wherever possible, only resorting to user decision or intervention when absolutely necessary.

The process of selecting a VID can be broken down into three steps. The first two steps are concerned with privacy policies, which are concerned with the access rights that a service may have to the personal data of the user. This is beyond the scope of this paper and will be described in detail in a subsequent paper.

If the first two steps are successful, they result in a list of one or more VIDs that can be selected for use with this service.

The third step uses the results produced by these two steps to select the actual VID to be used. In this step, a special type of preference rule, referred to as a *User VID Selection Preference* is used to select which VID should be used. User VID Selection Preferences define the circumstances under which a VID should be used and with what kind of service. The outcome of the evaluation of these preferences will state that a specific VID should be used in a specific situation.

This means that these preferences contain references to actual VID identifiers in contrast with other user preferences in which there are only references to specific context data. In the case of a new situation where a VID cannot be determined from preferences, the system should explicitly query the user at this stage and offer the list of VIDs for the user to choose from or allow him/her to create a new VID for this situation.

There is another type of privacy related preference rule which is used for context obfuscation. These preferences are used to determine the level of detail in which context items are delivered to services. These preferences only apply to data that can be obfuscated such as location and user activity.

A typical example of context obfuscation arises where a user may be prepared to release some information about their location but not in any detail. For example, a lecturer might be prepared to let students know that they are at university although not allow them to access their exact location. The privacy preferences for this may depend on the user's location, time of day/day of week, the service/user wishing to access his/her location, etc.

5. User privacy preference

The format of the privacy preference is currently under discussion and at present a range of industry standards is being considered (including XACML [19] and P3P [20]) as well as the possibility of creating custom privacy preferences.

The format of user preference rules for VID selection is straightforward. Conditions can include context conditions such as the location of the user, the current time, his/her activity and any other context attribute that exists in the context management system. The outcome specifies a specific VID to be used. There will be cases where no VID will match the user's VID selection preferences and in these cases, the user should be queried using a Graphical User Interface to select a VID from his/her pool of VIDs or be offered the option to create a new VID that will match in this case. If the latter is what the user wishes to do, the new VID will reference a list of user data and a VID selection preference will be set up for this VID to be used in the specific context in which it was created.

In the case of the context obfuscation preferences the format is much the same. The condition part is similar to that for user VID selection while the outcome determines whether or not the context obfuscation component is invoked to alter specific context data before they are disclosed to the service. Obfuscation preferences are retrieved and evaluated whenever a service requests a piece of user data that can be obfuscated and not during the process of VID selection.

6. Protecting, Creating and Refining User Preferences for Privacy

While user preferences in general represent sensitive data which needs to be protected from unauthorised access, user preferences for privacy are even more crucial. Although their format is essentially the same, the action performed is highly confidential since they affect the selection of VIDs. Thus this set of user preferences needs to be treated differently from the rest of the user profile.

One simple way of handling this would be to create a special-purpose preference management subsystem together with a learning component, which is a subset of the normal preference management subsystem, and which is contained completely within the Security and

Privacy subsystem. This would ensure privacy although at the expense of a considerable amount of duplicated code.

An alternative solution would be for the Security and Privacy subsystem to utilize the normal preference management and learning facilities of the pervasive environment even though these are not trusted. It can do so by using cryptographic techniques. By encrypting actions relating to the selection of VIDs before passing information to the preference management subsystem, and decrypting the information returned, the privacy of the user can be protected. The preference management and learning subsystem can handle the preferences as it does for any other service without understanding the actions. This solution avoids the expense of the additional code.

One area that is common for all preferences is that of the preference evaluation engine. By maintaining the same format of preferences for both purposes, the system can use a common engine to evaluate the conditions and offer the outcome to the appropriate component.

The first of the solutions is currently being investigated within Daidalos although the second approach may be explored later.

7. Conclusion

The problem of protecting the privacy of the user in a ubiquitous or pervasive environment is generally acknowledged as one of the most important issues that need to be solved if such systems are to be acceptable to the user.

In the Daidalos project a pervasive system is being developed in which a system of virtual identities (VIDs) is being used to hide the real identity of the user and thereby provide privacy protection through pseudonymity.

This paper discusses the challenges presented in providing adequate protection of privacy in the context of pervasive systems. For services to be context aware, personalized or simply “pervasive”, such a system must maintain large amounts of personal data and disclose these when required. This practice poses enormous threats to the privacy of individuals if not handled with the utmost care and protection.

The paper goes on to describe a solution that is being investigated to address these challenges in the context of the Daidalos pervasive system and for which implementation is under way.

8. Acknowledgment

This work was supported in part by the European Union as part of the Daidalos project under the Sixth Framework Programme and the PERSIST project under Framework 7. The authors wish to thank all colleagues in the Daidalos project developing the pervasive system, and especially those who have developed the concepts and components for handling privacy. However, it should be noted that this paper expresses the authors’ personal views, which are not necessarily those of the Daidalos consortium. They also gratefully acknowledge the support of the European Union for the Daidalos project but note that apart from funding the Daidalos project, the European Commission has no responsibility for the content of this paper.

9. References

[1] M. Weiser, “The computer for the 21st century,” *Scientific American*, vol. 265(3), pp. 94-104, 1991.

[2] B. Warneke, M. Last, B. Liebowitz, and K.S.J. Pister, "Smart dust: Communicating with a cubic-millimetre computer", vol 34 (44), January 2001, pp. 2-9.

[3] M.J. Sailor, and J.R. Link, "Smart dust: Nanostructured devices in a grain of sand", *Chem. Commun*., 11, pp. 1375-1383, 2005.

[4] D. K. Arvind, “Speckled Computing,” in *Proc. Nanotech 2005*, Anaheim CA, USA, 2005, 3, pp 351-354.

[5] The UK Grand Challenges Exercise. Available: http://www.ukcrc.org/grand_challenges/

[6] M. H. Williams, N. K. Taylor, I. Roussaki, P. Robertson, B. Farshchian, and K. Doolin, “Developing a Pervasive System for a Mobile Environment,” in *eChallenges 2006 – Exploiting the Knowledge Economy*, IOS Press, 2006, pp. 1695 – 1702.

[7] Y. Yang, F. Mahon, M. H. Williams, and T. Pfeifer, “Context-aware Dynamic Personalized Service Re-composition in a Pervasive Service Environment”, in *Proc. 3rd IFIP Int. Conference on Ubiquitous Intelligence and Computing (UIC 06)*, Wuhan, China, Springer Verlag LNCS 4159, Sept. 2006, pp. 724-735.

[8] M.H. Williams, I. Roussaki, M. Strimpakou, Y. Yang, L. MacKinnon, R. Dewar, N. Milyaev, C. Pils and M. Anagnostou, “Context Awareness and Personalisation in the Daidalos Pervasive Environment”, in *Proc. Int. Conference on Pervasive Systems (ICPS 05)*, Santorini, July 2005, pp. 98 – 107.

[9] J. Porekar, K. Dolinar and B. Jerman-Blazic, “Middleware for privacy protection of ambient intelligence and pervasive systems”, *WSEAS Transactions on Info. Science and Applications*, vol. 4(3), pp. 633-641, 2007.

[10] T. Z. Zarsky, “Thinking Outside the Box: Considering Transparency, Anonymity and Pseudonymity as Overall Solutions to the Troubles of Information Privacy”, *Miami Law Review,* 58(4) 2004, p. 1301

[11] C. Kalloniats, E. Kavakli, and S. Gritzalis., “Dealing with Privacy Issues during the System Design Process,” in *Proc. 5th IEEE Int. Symposium on Signal Processing and Information Technology,* December 2005, Athens, Greece.

[12] A. Pfitzmann, and M. Köhntopp, “Anonymity, unobservability, and pseudonymity,” in H. Federrath, Editor, *Designing Privacy Enhancing Technologies* , 2001, pp. 1–9.

[13] A. Brar, and J. Kay, “Privacy and Security in Ubiquitous Personalized Applications,” in *Proc. User Modelling Workshop on Privacy-Enhanced Personalization*, Edinburgh, UK, July 2005.

[14] M. Langheinrich. “A privacy awareness system for ubiquitous computing environments”, in Proc. 4th Int. Conf. on Ubiquitous Computing, London, UK, 2002, pp. 237--245.

[15] J.I. Hong, and J.A. Landay, “An Architecture for Privacy-Sensitive Ubiquitous Computing”, in *Proc. 2nd Int. Conference on Mobile Systems, Applications, and Services (MobiSYS)*, Boston, Massachusetts, USA, 2004.

[16] A. Kobsa, and J. Schreck, “Privacy through pseudonymity in user-adaptive systems,” *ACM Trans. Internet Techn*., Vol. 3(2), 2003, pp. 149-183.

[17] J.R. Rao, and P. Rohatgi, "Can Pseudonyms Really Guarantee Privacy?", in Proc. 9th USENIX Security Symposium, Denver, Colorado, Aug. 2000.

[18] J. Girao, A. Sarma, and R. Aguiar, “Virtual identities - a cross layer approach to identity and identity management“, .in *Proc. 17th Wireless World Research Forum*, Heidelberg, Germany, November 2006.

[19] OASIS XACML homepage. Available at: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml

[20] The Platform for Privacy Preferences 1.1 (P3P 1.1) specification. Available at: http://www.w3.org/TR/P3P11/