User preferences to support privacy policy handling in pervasive/ubiquitous systems

User Preferences to Support Privacy Policy Handling in Pervasive/Ubiquitous Systems

Elizabeth Papadopoulou, Sarah McBurney, Nick Taylor, M. Howard Williams, Yussuf Abu Shaaban

School of Math. and Comp. Sciences, Heriot-Watt University, Riccarton, Edinburgh, UK

{ ceeep1, ceesmm1, nkt, mhw, ya37}@macs.hw.ac.uk

Abstract

An important approach for handling user privacy in ubiquitous or pervasive systems is identity management, in which the user has a number of different virtual identities that conceal his/her real identity. One extension of the basic approach identifies the private data needed by services and uses the notion of a privacy policy to determine what access should be granted to private data by a service. This can then be used to determine an appropriate virtual identity. However, this is fairly complex and difficult for a naïve user to set up and control. Thus a major challenge lies in determining to what extent the decisions relating to the selection of virtual identities can be done automatically, and to what extent the user needs to be involved. This paper describes an approach in which user preferences are used to assist in taking these decisions - both to generate privacy policies and to select an appropriate virtual identity. The user preferences are simpler for the user to create and modify and are also easier for automatic learning techniques to update. This approach will help to create more user-friendly and acceptable identity management systems. These ideas have been explored within the Daidalos pervasive system while further work is being carried out for the Persist system.

Keywords: *User preferences; privacy; pervasive systems; policies*

1. Introduction

The importance of security and privacy in ubiquitous and pervasive systems is universally agreed. This paper is an extension of initial work in this area that was presented in [1].

The original vision of ubiquitous computing portrayed an environment surrounding the user that is filled with computing entities, supporting the user in a variety of ways without continual direction [2]. Since then there have been significant developments in areas such as sensor technologies and communications that are bringing us closer to enabling these predictions to be realised. The problem for the user lies in the increasing complexity that needs to be managed. Pervasive computing seeks to address this and to enable the user to control and manage this situation [3]. Although a successful system has not yet emerged, there is a growing view that during the next decade or so acceptable solutions will be found for many of the outstanding problems facing ubiquitous and pervasive computing and by 2020 this technology will be a reality. In the meanwhile some major problems still lie ahead and some of the global challenges of the next decade lie in this area [4].

In order to develop a pervasive computing system that is acceptable to the end user, it is important that it should satisfy two end user requirements:

(1) It should take account of the user's needs and preferences in any relevant decision making. With any system as complex as this, it is essential that it should adapt its behaviour according to the individual needs and preferences of the end user. The importance of incorporating user preferences is generally accepted and has been recognized in a number of projects, where preferences are either entered manually by the user or where learning is used to support the acquisition of preferences.

(2) It should adequately protect the privacy of the user. Both users and services need to know which services they can trust and what information they can share with them. This starts with the identity of the user – and whether or not the user is prepared to reveal his/her real identity to a service. To handle this, a

system of virtual identities may be used. The other aspect of this is authorisation – deciding who should be given access to what. To handle this an extension of the basic idea of virtual identities requires the pervasive system to identify the private data belonging to a user (e.g., location, credit card details) that a service wishes to access and then determine what access should be granted to the service and under what conditions. This may be achieved through the use of *privacy policies*.

However, privacy policies are fairly complex and difficult for a naïve user to set up and control. This paper describes how these two requirements can be brought together to use user preferences to support this aspect of privacy. It describes how user preferences can be used to generate privacy policies and to select virtual identities. These preferences may have the same format as other user preferences in the system and are therefore simpler for the user to understand and hence to create and modify. They are also easier for automatic learning techniques to generate automatically and for the user to understand what has been produced by the learning system. Thus by building up a flexible context-aware set of user preferences that can be used to assist in taking these decisions, one can increase the degree to which this can be handled automatically, and improve acceptability by the user.

These ideas have been explored within Daidalos, a European research project, a major aim of which was to develop a pervasive system, focussing especially on mobile users, in which security and privacy are key components. This work is being continued in Persist, another European research project developing a pervasive system based on Personal Smart Spaces.

This paper outlines briefly the approach and how it is being implemented. The next section provides a brief background to personalisation and user preferences and to privacy and pseudonymity followed by a brief introduction to Daidalos and Persist. Section 3 concentrates on virtual identities and the process of selecting an appropriate one. It also outlines how user preferences and personalization can assist in the automatic selection of virtual identities. Section 4 describes user privacy preferences while Section 5 presents more detail on the formats of the user preferences for virtual identity selection. Section 6 discusses some research challenges for the future, while Section 7 provides a brief conclusion.

2. Background

This section describes some of the background relating to this research. It gives a brief overview of issues relating to personalisation and to privacy, and then describes briefly the two projects, Daidalos and Persist.

2.1. Personalisation and User Preferences

The importance of individual user preferences and their application in decision making in a ubiquitous environment is generally accepted. In such an environment *personalization* refers to the process of creating, maintaining and applying user preferences in decision making since it has the effect of tailoring the system's behaviour to the individual needs and wishes of the user so that it appears or acts differently for different users or for the same user under different circumstances.

Thus far the work done on different ubiquitous/pervasive systems has incorporated personalization techniques with varying degrees of success. Early developments concentrated on the use of context information rather than on user preferences – producing a context aware rather than a personalized approach. However, the importance of incorporating user preferences into the decision making was soon identified and projects such as the Intelligent Home [5] and Blue Space [6] implemented both context awareness and personalization - although they relied on user input of preference information, resulting in minimal sets of user preferences.

The problem of capturing and maintaining user preferences was soon recognized and the need to assist the user in this process was established as an important requirement for future systems. As a result, projects such as the Adaptive House [7], GAIA [8] and MavHome [9] use monitoring and learning algorithms to gather preferences and environment information such as user movement and actions which are used to predict future movements and actions. Based on predictions, environments are automatically adapted by applying user preferences. However, removing the possibility for user input reduces user control which may lead to confusing or frustrating situations.

The Synapse project tries to find a balance between automation and user control by operating in active or passive mode [10]. If a preference has a probability above some threshold, it is applied automatically in active mode, whereas passive mode consults the user with suggestions before preference application. Although this can produce more accurate personalization, there is a risk that the user could be inundated by pop-up messages.

More recently, projects such as Ubisec [11], Spice [12] and Mobilife [13] aim to provide the user with

personalized services in a global environment. Once again preferences are applied automatically but improved personalization is provided by implementing more responsive implicit personalization mechanisms, which respond rapidly to changes in the user's behaviour patterns and update the set of user preferences in real time.

2.2. Protecting Privacy

Privacy can be regarded as "the right of individuals to protect their ability to selectively reveal information about themselves" [14]. 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 [15][16]. Pervasive systems have a lot in common with the Web and the same requirements apply.

A number of papers (e.g., [17][18][19]) 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 [14].

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 which place different restrictions on the information they release to services [17]. 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 state that anonymisation hides the relationship or linkage between an individual user and his/her stored personal data [20]. 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 [14]. 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 [21].

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 [21]. 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.

2.3. Daidalos and Persist

Daidalos is a large European research project, whose overall aim was to create a pervasive environment for mobile users [22]. This was achieved by integrating a range of heterogeneous networks and devices and creating a pervasive system on top of this which protects the user from the complexity of the underlying infrastructure while providing personalized and context aware services with minimal user intervention.

Within Daidalos a layered approach was adopted, separating the lower level network functionality from the higher level pervasive system. Security and privacy were a priority area in Daidalos and affected all components.

At the higher level the pervasive system depended on personalization and user preferences. Initially a simple approach was used to capture user preferences but later this was extended to use both stereotypes and learning to capture the preferences effectively [23].

These preferences are used for Service Discovery and Selection, Service Composition, Session Management, Security and Privacy, Context Management and Personalization [24]. In addition, user privacy is essential, and this functionality affects both knowing who is running what services and controlling access to user data.

On the other hand, the Persist project is a much smaller project, funded under the Seventh Framework. It builds on some of the work initiated in Daidalos. Once again it aims to produce a prototype pervasive system, but this one will be based on the notion of a self-improving Personal Smart Space (PSS). The vision of Persist is that a Personal Smart Space will replace the fixed smart spaces associated with buildings and the mobile ad hoc networks associated with users, creating a single uniform approach. This will provide an interface to link users to the various services and devices that surround them, as well as to other neighbouring Personal Smart Spaces.

The notion of a smart space is usually associated with a real physical space. From this point of view a smart space can be defined as "*a multi-user, multi-device, dynamic interaction environment that enhances a physical space by virtual services*" [25][26]. The services are the means of interaction between participants, objects and the smart spaces. A Personal Smart Space extends this notion but is not necessarily associated with a fixed location. It is a collection of devices connected in an ad hoc network that may be fixed in a particular location (e.g., a room, office, house, etc.) or may move around with the user. It may even be composed of devices located in different locations but associated with the same user. This type of architecture has some significant advantages over the more conventional approaches.

3. Using User Preferences to Select Virtual Identities

Pseudonymity is achieved in Daidalos through the use of multiple Virtual Identities (or VIDs) [27]. These VIDs 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 VIDs can be linked to any of the others so that if a user uses two VIDs with the same service, that service will treat these as two different users. 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. This also allows for good personalization practices since users can use services for different activities and have different preferences for each activity.

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.

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 assumption is too simplistic and puts an unnecessary burden on the user. In particular, as the user accumulates VIDs, this will become increasingly arduous, just as remembering different user names and passwords for different Internet sites has become a problem. The situation is further complicated by the fact that the use of different VIDs may depend on the context of the user. For example, the user may have two different VIDs for the same service, one associated with work use and the other for use at home. It is essential, therefore, that the system itself should manage the automatic selection of VIDs wherever possible and only require action from the user when it cannot take a decision or when the user disagrees with the decision taken. This is important in order to realise a user-friendly pervasive environment that is acceptable to the user.

A VID is more than just a user name but also defines the set of user data that can be made available to a service. Consequently before selecting a VID one needs to establish what data the service will want to access. Then, if the system is not willing to provide all the access requested by the service, or there are any

constraints that the user has, it needs to negotiate with the service and reach an agreement about the access rights that will be granted to the service and the conditions under which these are granted.

The user's wishes with regard to access to personal data items need to be formalised in an appropriate way. Thus for each item of personal data any constraints that the user may want to place on access to it are captured and expressed in the form of a Privacy Policy. The latter is used as the basis for negotiating with the service and this process of negotiation is referred to as Privacy Policy Negotiation (PPN). If an agreement can be reached the result is referred to as a Privacy Policy Agreement. Once this has been agreed an appropriate VID can be selected. Thus the whole process can be broken down into four steps as follows.

(1) *Establish requirements for data*. The most important factor in determining the VID to be used in any particular situation is the user data that a service needs to access and the access rights (read, write, update) it requires. Thus when a service requests a VID, the first step is to determine what user data the service will want to access. This can then be compared with the user's instructions on access to his/her data as expressed in the privacy policies. This specifies what access should be given to any item of data and under what conditions this access may be granted – for example, the length of time this data may be held or whether or not it may be shared with other services.

(2) *Negotiate use of data*. Once the data requested has been matched against the privacy policies, the system may need to negotiate with the service to ensure that whatever user data is provided to the service will only be used in accordance with the user's wishes. To do this the set of privacy policies is passed to the Negotiating Agent to negotiate with the service on behalf of the user the terms of use based on these outcomes. This negotiation should result in an agreement that meets all the requirements in the privacy policies. This process is similar to that of trust negotiation [28].

(3) *Match PPN outcomes with potential VIDs.* Whether or not the Negotiation Agent needs to conduct a negotiation with the service, the system establishes a Privacy Policy Agreement which consists of a list of private data items (such as personal information, context attributes or preferences) that the service can access. This list is then used to identify the set of possible VIDs that will allow access to all of the items in the list without providing access to much else of significance.

This should result in the identification of one or more VIDs that can be selected for use with this service. If no VIDs are found which would provide the required access to the data items in this list, the user needs to be consulted. A GUI is invoked and the user is given the option of changing the data access constraints associated with an existing VID or creating a new VID with the required data access properties. Alternatively the user could select a different service as there is no way in which the current service can be run with the VIDs that are available.

(4*) Select final VID.* Once the set of possible VIDs has been established, the final VID can be selected from this list. This process happens even when only one matching VID is found.

4. User Privacy Preferences

In order to make use of user preferences in this process, the approach described in the previous section can be extended with an additional step at the beginning of the process in which user preferences are used to generate the privacy policy. One way of viewing this is to divide step (1) as follows:

(1a) *Establish data item requirements*. Here the pervasive system needs to determine from the service what items of private data it wishes to access.

(1b) *Generate privacy policies*. From the user preferences relating to these particular data items, generate a set of privacy policies.

For this purpose one may have a set of preferences, referred to as *User PPN preferences* that define what the user wishes in each situation. These may depend on external factors such as context conditions (e.g., the user's location, activity, people in his/her proximity, etc) or service-specific conditions (e.g., reputation of service). To aid the user privacy preferences, service trust levels are maintained in the system for each user. Some users may believe that service X is sufficiently reliable that it can be trusted with confidential personal information while others on the other hand might not trust it and only be prepared to provide access to limited subsets of information (or possibly even none at all). To aid the user, the system can maintain service trust levels for each user. These can be used as part of a condition in a user PPN preference. In each case, the result of evaluating a PPN preference tells the system whether or not a piece of personal data can be allowed to be disclosed, and under what conditions. The evaluation of these PPN preferences for all the requested user data results in a set of privacy policies.

This set is used by the Negotiating Agent to negotiate with the service on behalf of the user the terms of use based on these outcomes.

One can also use user preferences in the final step to select the actual VID. User VID selection preferences are used to determine which VID to use. 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.

Thus two different types of preference rules may be used to support the user in the process of selecting a VID, namely *User PPN preferences* and *User VID Selection Preferences*. In the Daidalos system, there is also a third type of privacy related preference which is used for "context obfuscation". These are used in the case of certain context attributes where the level of detail that is returned may be controlled. For example, in the case of a request for location information, if one is in one's office at university, one might return the office number or one might return the building or possibly just the university or even the part of the city or the city itself. Thus these preferences determine the accuracy to which context items are delivered to services. However, they are beyond the scope of this paper.

5. Formats of User Privacy Preferences

The format of the privacy policies is based on the industry standards P3P [29] and XACML [30]. It also has the facility for users to create their own custom privacy preferences.

On the other hand the formats for the user PPN preference rules are the same as those adopted for all preference rules in the Daidalos system so that they can be easily understood and manipulated by the user, a basic requirement in designing privacy aware systems [31]. This consists of a simple "if-then(-else)" rule in which the condition part is a Boolean expression comprising one or more simple conditions such as checks on context attributes, the status and attributes of other services being run by the user, attributes of the service requesting access to the data, previous usage of the requesting service, trust levels of the service which have been formed by the user and/or groups of users sharing such trust levels. Each then-part or else-part may be either an action or a nested if-then(-else) statement. This has been fully specified but constraints on space do not permit a fuller discussion on this here.

The action part of a user PPN preference rule contains a list of the conditions that govern the disclosure of a piece of user data to a service. Thus the outcome after evaluating such a preference would be either positive (i.e. disclose the piece of data), negative (i.e. do not disclose it) or a conditional expression of the form "positive if a list of requirements is met". These latter requirements are conditions such as the data retention policy of the requesting service, the data usage policy of the requesting service and other such conditions subject to negotiation with the service.

A PPN preference rule is associated with a single data item. Thus for each data item or context attribute associated with a particular user, one may have a separate rule. To illustrate this consider the following example:

IF symbolic_location = 'work' AND
dayOfWeek = 'weekday' AND
LocalTrustLevel(requestor) >0.5 AND
GlobalTrustedReputationLevel(service) > 0.7
THEN PrivacyPolicyRule:
Effect: "Permit"
Obligations:
1) Data_Retention_Policy<12 hours
2) Share information with 3rd parties: NO

The above example of a PPN preference in our format is a rule with four conditions. The first two conditions are context conditions which specify that the rule should be enforced only when the user's symbolic location (in a high level form inferred from raw sensor data such as GPS coordinates, as opposed to the original raw data) is "work" and the current day of the week is a weekday. We are aware of the complexity associated with inferring symbolic location from the raw location data but this is dealt with by others within the Daidalos system and is beyond the scope of this paper.

The following two conditions require the use of other components in the system such as the local trust handling component and the global reputation system. In the first case, the local trust component refers to a

system that gathers information about previous transactions with services by monitoring the use of the service by the user and prompting the user to provide such information. Each service is assigned a trust value within some range of values to indicate the level of trust the user has in the privacy practices of that service. In the case that a service has not been used before, the local trust component can deduce a range of trustworthiness for a service by examining the trustworthiness of services with similar privacy practices and guarantors. In the example given, the preference states that the local reputation value should be no less than 0.5. This value, the service reference and a number of parameters pertaining to the service are given to the local trust system to evaluate whether it is true or not.

The fourth condition refers to a global reputation system that a user can query to acquire the trustworthiness level of a service as viewed by a collection of users. This value has been calculated by combining the trust values submitted to that system by a number of different users. Thus when this condition in the preference rule is evaluated, a query is forwarded to one or more reputation systems to acquire the trustworthiness of this service as judged by other users. The values returned are averaged based on an algorithm that takes into account the user's indicated trust in these reputation systems. The final value is checked against the value stated in the preference (in our example the user has indicated the value of 0.7).

The outcome of a PPN preference specifies what should happen if the overall condition is met. In the example, the preference states that the service should be permitted to access the specific piece of data if and only if a number of requirements are met by the service itself. The outcome of a PPN preference is a collection of requirements that the system will negotiate with the service. If the service does not agree with these requirements during the negotiation process, the data item will not be disclosed to the service. In the example, two requirements are specified for the service to agree to. The first declares that the data item value should not be logged by the service for more than 12 hours since the first initiation of the service. This is required so that services do not accumulate a large history of the user's context information that can result in compromising his/her privacy. The second requirement states that any disclosed information will not be forwarded or sold to third parties by the service.

It should be noted that it is possible that more than one preference can exist that determines whether or not a specific piece of data can be disclosed. For example, the user can set a preference whose effect is to deny disclosing his/her location if certain conditions are met. Depending on the conditions specified in the preferences, it is possible for the system to result in two outcomes, one that allows the disclosure of the information and one that does not. In this case, the system will not allow the disclosure of the information because a rule with effect "Deny" has precedence over all preferences with effect "Permit".

If the conditions are met, the outcome will be translated on the fly to the following XACML policy snippet:

```
    <Obligation ObligationID=”data_retention”
FulfillOn=”Permit”>
    <AttributeAssignment AttributeId=”data_retention”
DataType=
    ”http://www.w3.org/2001/XMLSchema#Duration”
>
    P0Y0M0DT12H0M0S
    </AttributeAssignment>
    <Obligation ObligationID=”3rd_pty_disclosure”
FulfillOn=”Permit”>
    <AttributeAssignment
AttributeId=”3rd_pty_disclosure” DataType=
    ”http://www.w3.org/2001/XMLSchema#String”>
    NO
    </AttributeAssignment>
```

In practice the condition part will in general be more complex than this, growing as the user adds to it or as automatic learning modifies it. To make things easier for the user, the notion of a "situation" has been introduced, which the user can associate with a specific set of context values.

When the PPN preferences for the whole set of data requested by the service are evaluated and the outcomes are combined, the result is a privacy policy set that specifies under which circumstances access to user data should be granted and forms the basis for negotiation with the service. This negotiation should result in an agreement that meets all the requirements in the privacy policy set. Thus the outcome of the negotiation is a privacy policy that specifies under which circumstances access to user data should be granted.

The user VID selection preferences have the same basic format except that the action part specifies a VID to be used. Thus 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. For example, the user can have different VIDs for a VoIP application depending on his/her activity, location and current time:

```
IF (location='work' OR time.between(0900,1700))
AND dayOfWeek='weekday'
THEN VID='workVID'
ELSE VID='defaultVID'
```

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.

6. Some Research Issues

Some research challenges associated with this approach include the following. The first challenge concerns the way VIDs are handled to ensure VID isolation. A fundamental assumption here is that no service should have access to more information on a user's VIDs than is absolutely necessary for its functioning. In particular, no service should be able to associate independent VIDs belonging to the same user, and hence infer or gain access to personal information on the user to which it is not entitled. This applies to all services outside the Security Manager module and to some extent even within it. This has consequences for the design of the user preference subsystem.

Another major issue is how to engage the user in the decision making. If it is completely automatic, it will be difficult for the user to change when the need arises; if it is completely manual, it will be too arduous for the user. A compromise is to take the decision for the user and inform him/her of the VID selected, giving him/her the opportunity to intervene and change the VID selected. If the user does so, he/she may change to an existing VID or create a new one. The design of suitable GUIs with flexible representations of VIDs without releasing more information than necessary and enabling VID linkage is another serious challenge.

By monitoring the user's actions in accepting or changing VIDs and applying machine learning techniques to this information, the set of user preferences can be built up and maintained automatically. However, the problem of VID isolation affects the machine learning and could result in a laborious process - although this has been overcome in our system. Nevertheless one is still faced with the problem of distinguishing between short-term and long-term changes in preferences.

At a different level one has issues relating to the storage and protection of the preferences. While user preferences in general represent sensitive data which needs to be protected from unauthorised access, user preferences for privacy are even more crucial because of the way in which they are accessed and used and because they control the degree of access a service is permitted to the user's personal data (including other preferences). 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.

7. Conclusion

In developing pervasive computing technologies that are acceptable to the end user, it is essential to take account of user needs and preferences to personalize decision making within such a system. One important area where they may be used to improve the user-friendliness of pervasive systems is in the identity

management approach to dealing with user privacy. User preferences can be used both in determining what information can be released about the user and in the process of selecting a virtual identity to hide the real identity of the user. These are easier for the user to understand and manipulate, especially if the formats of such user preferences are consistent with those of other user preferences in the system.

One of the aims of the Daidalos project was to develop a pervasive system which uses a system of virtual identities (VIDs) to hide the real identity of the user and thereby provide privacy protection through pseudonymity. At the same time a lower-level objective was the provision of different forms of personalisation through the use of user preferences to make the system acceptable to an end-user. To this end the approach described in this paper was developed.

The Persist project is another European research project aimed at developing a pervasive system based on a radically different approach – the notion of Personal Smart Spaces. The approach described here will be used within the Persist prototype.

This paper is concerned with the use of virtual identities in providing adequate protection of privacy in the context of pervasive systems. It extends and elaborates on the ideas presented in [1].

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 has been investigated to address these challenges in the context of the Daidalos pervasive system and which will be used in the implementation of the pervasive system prototype in the Persist project.

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.

References

[1] E. Papadopoulou, S. McBurney, N. Taylor, M.H. Williams, K. Dolinar and M. Neubauer, "Using User Preferences to Enhance Privacy in Pervasive Systems", Third Int. Conf. on Systems (ICONS 2008), Mexico, 2008, pp. 271-276.

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

[3] M. Satyanarayanan, "Pervasive computing: vision and challenges", *IEEE PCM*, vol. 8(4), pp. 10 - 17, 2001.

[4] The UK Grand Challenges Exercise. Available: http://www.ukcrc.org.uk/grand_challenges/ 28.05.2009

[5] V. Lesser, M. Atighetchi, B. Benyo, B. Horling, A. Raja, R. Vincent, T. Wagner, P. Xuan and S. Zhang, "XQ.: The Intelligent Home Testbed", in *Proc. Anatomy Control Software Workshop (Autonomous Agent Workshop)*, 1999, pp. 291-298.

[6] S. Yoshihama, P. Chou and D. Wong, "Managing Behaviour of Intelligent Environments", in *Proc. First IEEE Int. Conf. on Pervasive Computing and Communications (PerCom '03)*, 2003, pp. 330-337.

[7] M. C. Mozer, "Lessons from an Adaptive House", in D. Cook & R. Das (Eds.), *Smart Environments: Technologies, protocols and applications*, 2004, pp. 273-294.

[8] B. D. Ziebart, D. Roth, R. H. Campbell and A. K. Dey, "Learning Automation Policies for Pervasive Computing Environments", in *Proc. 2nd Int. Conf. on Autonomic Computing (ICAC '05)*, 2005, pp. 193-203.

[9] M. G. Youngblood, L. B. Holder and D. J. Cook, "Managing Adaptive Versatile Environments", in *Proc. 3rd IEEE Int. Conf. on Pervasive Computing and Communications (PerCom '05)*, 2005, pp. 351-360.

[10] H. K. Y. Si, "A Stochastic Approach for Creating Context-Aware Services on Context Histories in Smart Home", in *Proc. ECHISE2005, Pervasive '05*, 2005, pp. 37-41.

[11] J. Groppe and W. Mueller, "Profile Management Technology for Smart Customizations in Private Home Applications", in Proc. 16th Int. Workshop on Database and Expert Systems Applications (DEXA '05), 2005, pp. 226-230.

[12] C. Cordier, F. Carrez, H. Van Kranenburg, C. Licciardi, J. Van der Meer, A. Spedalieri, J. P. Le Rouzic, and J. Zoric, "Addressing the Challenges of Beyond 3G Service Delivery: the SPICE Service Platform", in *Proc. Workshop on Applications and Services in Wireless Networks (ASWN '06)*, (2006).

[13] M. Strutterer, O. Coutand, O. Droegehorn, and K. David, "Managing and Delivering Context-Dependent User Preferences in Ubiquitous Computing Environments", in *Proc. Int. Symp. on Applications and the Internet Workshops (SAINTW '07)*, 2007.

[14] 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

[15] 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.

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

[17] 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.

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

[19] 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.

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

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

[22] 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.

[23] E. Papadopoulou, S. McBurney, N. Taylor, M. H. Williams and G. Lo Bello, "Adapting Stereotypes to Handle Dynamic User Profiles in a Pervasive System", in Proc. 4th Int. Conf. on Advances in Comp. Sc. and Tech. (ACST '08), 2008, pp. 7-12.

[24] 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.

[25] T. Kindberg, et al, "People, Places, Things: Web Presence for the Real World", in *Proc. Third IEEE Workshop on Mobile Computing Systems and Applications*, 2000, 19-28.

[26] C. Prehofer, J. van Gurp and C. di Flora, "Towards the Web as a Platform for Ubiquitous Applications in Smart Spaces".

[27] 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.

[28] T. Yu, M. Winslett and K.E. Seamons, "Supporting structured credentials and sensitive policies through interoperable strategies for automated trust negotiation", *ACM Trans. Inf. Syst. Secur*. Vol. 6(1), pp. 1-42, 2003.

[29] The Platform for Privacy Preferences 1.1 (P3P 1.1) specification. Available at: http://www.w3.org/TR/P3P11/ 28.05.2009.

[30] OASIS XACML homepage. Available at: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml 28.05.2009

[31] 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.