# Security and Privacy in a Pervasive World

J. Clarke,
S. Butler
*LAKE Communications*
*Jim.Clarke@lakecommunications.com*
*Stephen.Butler@lakecommunications.com*

C. Hauser,
M. Neubauer,
*University of Stuttgart*
*hauser@ikr.uni-stuttgart-de*
*neubauer@ikr.uni-stuttgart,de*

P. Robertson
*German Aerospace Center, DLR*
*Patrick.Robertson@dlr.de*

I. Orazem,
A. Jerman Blazic
*Security Technology Competence Centre SETCCE*
*igor@setcce.org*
*aljosa@setcce.org*

H. Williams,
Y. Yang
*Heriot-Watt University*
*mhw@macs.hw.ac.uk*
*ceeyy1@macs.hw.ac.uk*

## Abstract

*DAIDALOS is a European project funded under the Sixth Framework, which aims to integrate a range of heterogeneous networks and to develop pervasive systems on top of them to provide the user with transparent access to personalised communication and information services in a complex environment. Security and privacy are key to the development of such a system. This paper addresses some major questions on how to build trust and confidence in complex environments of pervasive systems where many entities play different roles, mostly for some limited amount of time, and where a significant amount of personal related data travel through the network to deliver highly personalised services.*

## 1. Introduction

**DAIDALOS**, which stands for **D**esigning **A**dvanced **I**nterfaces for the **D**elivery and **A**dministration of **L**ocation independent **O**ptimised personal **S**ervices, is a project within the European Commission's Sixth Framework Information Society Technologies Programme. With around 46 partners (the exact number is a function of time), the main research effort is divided between a number of work packages. More details on Daidalos can be found on the web site www.ist-daidalos.org. One important thread that runs through the different research areas is that of security and privacy.

Another focus of the research in Daidalos is the development of pervasive systems. Thereby, Daidalos aims at a platform to allow third party providers to easily deploy new pervasive systems. The supporting services, Daidalos provides are called "enabling services" in contrast to the services provided by third parties using the Daidalos platform. To illustrate our interpretation of a pervasive service, imagine that you are walking along, watching a video on a hand held device. You enter a room with a large public display, and the video is automatically switched to the large display, which transparently charges you a couple of cyber-cents. This is an example of a pervasive service. From the Daidalos perspective, a pervasive service is regarded as "a context-aware composition of third party services and enabling services cooperating in a seamless manner, so that the user can focus on the content, without being bothered by the complicated technology that is used for achieving his/her goal" [1]. However, this simple example does raise a number of important questions. How does the system know that you are entering the room? How does it know that you want to use a large display whenever available? How does it know your charging account? How does it know whether you are authorised to use the large display? How does it know that the use of the display does not reveal any sensitive data?

Questions like this raise a crucial issue of pervasive systems – privacy and security. First of all, it is obvious that such a system must have access to confidential personal information about its users – because it is about to always act adaptively according to the user's personal situation. Thus, this information must be handled with care and protected in order to protect the user's privacy. For this, it is also necessary to provide for security of the system.

For achieving privacy, users must be aware of how personal data are processed and for which purpose personal data submitted are used. Furthermore, personal data collecting and processing must follow formal and legal scope. Users must be confident that they are interacting with genuine providers and not an unreliable third party. As one goal of pervasiveness is to minimise the user's interaction necessary to control the system, Daidalos strives for automation of this process as far as possible. Of course, it must also be ensured that no eavesdropper can look into communicated data, particularly in data that holds user personal information.

This brings us to the second important property of pervasive services – their dynamic nature. While the user is on the move, there will always be new services appearing in the vicinity of the user. Moreover, the actual network a user is associated with changes quite often. In addition, it is a widely acknowledged fact that future telecommunication systems will be opened to many providers – network providers, service providers and content providers. Thus, a user will frequently be confronted with new providers and will have to be authenticated by each of them. Compared to today's systems, this bears a huge challenge. Today, trust relationships are static and security settings require significant effort to configure. In a dynamic pervasive system with mobile users – acting also as providers – this is no longer appropriate.

Thus, Daidalos evaluates and develops possibilities for providing open pervasive systems while assuring protection of the user's privacy. Thereby, the wheel is not reinvented, i.e., Daidalos does not focus on problems already present in conventional systems as they are (and were already) addressed in other projects. Instead, it is focussed on the new challenges imposed by Daidalos, i.e., open systems and pervasiveness. Moreover, the goal of this paper is not to present a security and privacy evaluation of pervasive systems. Rather it is focussed on three innovative building blocks that will ensure privacy in pervasive systems. The general approach our work is based on is a multiple identities environment. In the context of each identity, it is possible to control the amount of disclosed personal information. This is feasible as a service does not need the full range of the user's sensitive context but only a selected portion of it.

The Daidalos concept follows examples of the real world and translates them into pervasive systems. A user of a pervasive system must be able to understand the purpose of personal information being collected at any time. When a new service is first encountered, a privacy policy must be negotiated bilaterally with the service prior to the use – this negotiation process is defined by personal rules. This privacy policy is the basis for an identity of the user's choice, which the service can then use to interact with him/her and access his/her data. Moreover, it is used to configure the access rights to the personal context information. These three building blocks – privacy policy negotiation, identity management and access control – take the burden of understanding all privacy impacts and doing the necessary configurations manually away from the user and thus supporting pervasiveness of the system.

To illustrate another aspect of the multiple identity functionality, consider the following situation. John is a doctor who works in general practice. Outside of work he has various roles within the community – e.g., he is a member of a local church committee, he is a keen golfer, etc. He also has a family life with a wife and two children. Thus John has a number of different facets to his life and chooses to use different identities for each of these so that the system can respond differently to his needs in different circumstances.

As a result when John is in his doctor role (using his doctor identity) he has a certain set of user preferences associated with this identity. For example, he may have certain preferences for QoS, cost of services, etc., as well as preferences like how to redirect any communications sent to him in this mode if he cannot respond to them. He may also have a particular set of people who are authorised to access his location. He may have authorisation to access particular resources, which are not available to all users, such as certain patient records in the hospital database.

On the other hand when John is in family mode (using his family identity) he may well have a different set of user preferences. For example, in doctor mode he may want the best QoS possible at any price, whereas in family mode he may have limits on what he is willing to pay. The set of people who are authorised to access his location will be different – geared around family and friends rather than work colleagues and superiors. He will have access to a different set of resources. For example, he no longer has access to the hospital database but does have access to the family calendar. And so on.

The same may be true of his roles in the community with different sets of preferences, access to different resources, etc.

In order to help John use these services easily, the system may provide automated support when preparing the (virtual) identities associated with his different roles. In particular, configuration of the system to handle all circumstances can be laborious. Based on what the system knows about the user's roles and their differences (e.g., professional vs. private focus) it might be able to negotiate different configurations in terms of which context it will reveal to other users and under which conditions, thereby reducing the configuration burden. The system also helps John by providing appropriate privacy protection. An important goal to achieve is to minimise the effort that John needs to devote to specifying the security configuration. As much as possible should happen behind the scenes, although John must still retain the capability to manage and decide manually. For novices, it is also possible to let experts define the privacy rules, while strictly following general formal (legal) rules. The system then will seamlessly act according to them.

The structure of the paper is as follows. In the next section, the objectives of the Daidalos project are shortly presented. Then, the identity concepts of Daidalos are presented followed by a description of the processes of privacy policy negotiation and the choice of identity. The paper finishes with a summary and a discussion of the approach.

## 2. Daidalos Objectives

The overall objective of **DAIDALOS** is to develop and demonstrate an open architecture based on a common network protocol (IPv6) to:

- Design, prototype and validate the necessary infrastructure and components for efficient distribution of services over diverse network technologies beyond 3G,
- Integrate complementary network technologies – ranging from the area of telecommunications to the area of broadcast – to provide pervasive and user-centred access to these services,
- Develop an optimised signalling system for communication and management support in these networks,
- Demonstrate the results of the work through strong focus on user-centred and scenario-based development of technology.
- Throughout all these parts of the project, security and privacy of all stakeholders – users, operators and providers – is to be protected.
- Support users in initially understanding and specifying their levels of privacy and reducing the overall effort and interaction due to privacy implications while using the Daidalos platform.

## 3. Identity Concepts

Daidalos is bundling applications and network services and making this transparent to users. For this, a consistent security and privacy framework based on multilateral security is being developed. For privacy purposes, the link between a user's identity and his/her personal information, e.g., context information must be concealed. However, for multilateral security, users must be accountable for their actions; this necessitates the use of pseudonyms rather than anonymity (see [3] for terminology proposal). In Daidalos, we refer to this pseudonym together with the information disclosed in the context of this pseudonym as a Virtual Identity (VID), because it is a kind of identity consisting of an artificial name of the user augmented by a set of attributes under which the user appears in the system. This set of personal data is controlled by a so-called Privacy Agent, which acts on behalf of the user and is controlled by him/her implementing the right of informational self-determination. As the set of personal data being revealed in a specific VID does not contain all information attributed to a user it resembles a restricted view on the user at the other side. Because this view is controlled and restricted as well as only present in the technical system we refer to it as **Virtual** Identity.

This set of information is a partial view on the user's overall context, which reflects his/her current situation, static personal data and preferences. In accordance with the principle of data minimisation of the European Parliament and of the Council (see [4] and [5]), the set of personal data disclosed to a service should exactly match the amount of data the service needs. Thus, the multiple identity approach follows and supports the principle of data minimisation and is a means for implementing this legislative requirement.

For accountability and charging purposes, Daidalos introduces a so-called Registration ID (RegID), which describes the user's attributes resulting from his/her registration. Typically, the registration is done with the provider of Authentication, Authorisation, Accounting, Auditing and Charging (A4C). This registration describes, e.g., which services he/she can use, how many Virtual Identities he/she can use in total and/or at the same time, and whether he/she is a premium or a standard user. Further, the account to be charged is specified. These Registration IDs serve multilateral security, because on the one hand they guarantee that the service provider will get the money for the service provided. On the other hand, based on the logging of service usages in combination with the used RegID, the user can verify (audit) the electronic receipts. Thus, non-repudiation requirements of both users and service providers will be met. The RegID usually is only known by the A4C provider, which acts as a trusted third party. Other providers – possibly including network operators – only see Virtual Identities of the user. It must be noted, that Daidalos' privacy & security concepts build on an open framework, where roles and actors may change or swap. While the A4C provider will usually serve as a focal point regarding the user's real identity – for charging purposes – the roles may also be distributed by introducing trusted third parties. Thus, it can also exist a third party knowing the user's real identity for clearing and the A4C provider only knows a pseudonymous VID.

## 4. Privacy Architecture and Processes

As in the real world, the world of pervasiveness has rules on how personal data should be handled. It is quite natural that the personal privacy data protection criteria of one user will be different from those of another one. They will depend on the degree of trust that the user has in the "neighbourhood" or from his personal principle manner [6], [7], [8]. Thus, when two parties would like to cooperate, they first need to agree on terms and conditions of personal data exchange and usage. In the overall Daidalos Security and Privacy Management (SPM) subsystem, the Privacy Policy Negotiation (PPN) component is responsible for this task. It is part of the Privacy Agent – one of two agents in the SPM. For this purpose the module compares the subjects' Privacy Policy Statements (PPS). As result there will be a privacy policy agreement as a list of exchanged and agreed statements. The statements contain issues such as which data may be disclosed, for which purpose and so on.

Privacy Policy Statements play a leading role in the user's personal data protection [9], [10]. The Privacy Policy Negotiation module unveils formal and technical terms of data processing, which also serve as an agreement between parties for potential audit trails. Security mechanisms such as encryption serve as underlying means preventing disclosure of data during and after the process. Privacy protection is achieved through a set of formal negotiation processes and the according actions taken on that result – selection of an appropriate identity and configuration of the access control to personal data.

Figure 1 shows the principle scenario of a service use. After discovering the wished service, the service composition component (SD/SC: Service Discovery / Service Composition) initiates the Privacy Agent inside the subcomponent for security and privacy management of Daidalos requesting a VID for this specific service use.

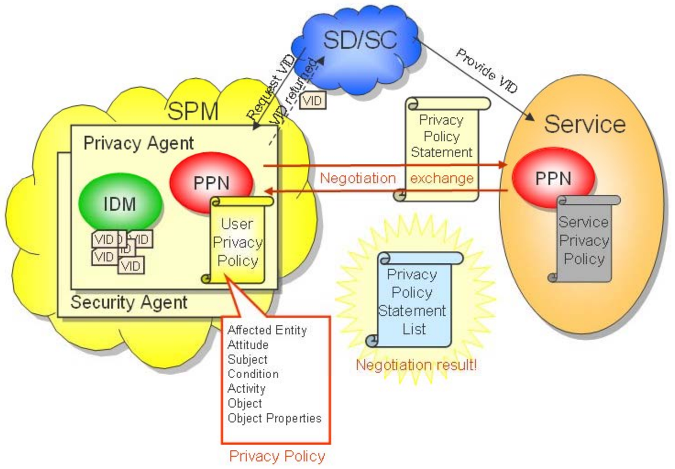

**Figure 1: Privacy policy negotiation**

This in turn triggers Privacy Policy Negotiation, handling the process of agreeing on the terms and conditions of personal data use in the context of this service use. The result of negotiation comes as a privacy policy agreement in a form of a Privacy Policy Statement. This intermediate result is used to create (or reuse) one of the user's virtual identities selected by the Identity Management (IDM) which will be described below. The virtual interpretation of a user may serve for more than one scenario and may be reused at any time.

Daidalos also addresses the problem of privacy policy management. It is hard to imagine that a user will manage his/her privacy conditions dynamically and use the services offered seamlessly at the same time.

Privacy policy management must therefore incorporate some user friendly interfaces and provide protection at the highest level with minimal user effort in a pervasive world.

Based on the negotiated privacy agreement, the Identity Management selects or creates a VID to be used for this service invocation. Therein, the Identity Management does not need to be aware of the actual values of personal data, e.g., context information to be disclosed. Instead, IDM uses a formal description of the type of context, e.g., facts like "in the context of VID 1 the location has to be revealed". This context description is part of a more general context ontology. Thus, it is possible to reflect the coherences between different pieces of context to also consider inference of sensitive data from originally unclassified data like shown in [2].

Based on this formal description, the Context Manager – a Daidalos platform component providing the actual values and the context ontology itself – creates unique identifiers for each piece of context information. These Context Identifiers (CIDs) are then used in Virtual Identities to reference the real value and to make it accessible to other users and services. Thus, a service will access, e.g., the actual value of the location of VID 1 by requesting the respective CID from the corresponding Context Manager. This CID is defined in form of a URL and comprises:

- The name of the host running a Context Manager instance of the overall distributed Context Management subsystem,
- A pseudonym of the Virtual Identity of which the context information should be retrieved,
- A description of the context information itself[1], e.g., "location", "temperature" etc.

Note, that the CID is not used for controlling access to context information. In order to prevent unauthorized access to personal information by guessing a CID or a brute-force attack, the Context Manager implements an Access Control Decision as well as an Access Control Enforcement point which are parts of the Security Agent inside the security and privacy management subsystem. As the IDM is in charge of controlling which personal data is revealed in which context to whom, IDM is also responsible for triggering the creation of access rights at the different Context Manager instances.

The software architecture of the IDM in the Daidalos prototypical implementation is depicted in Figure 2.

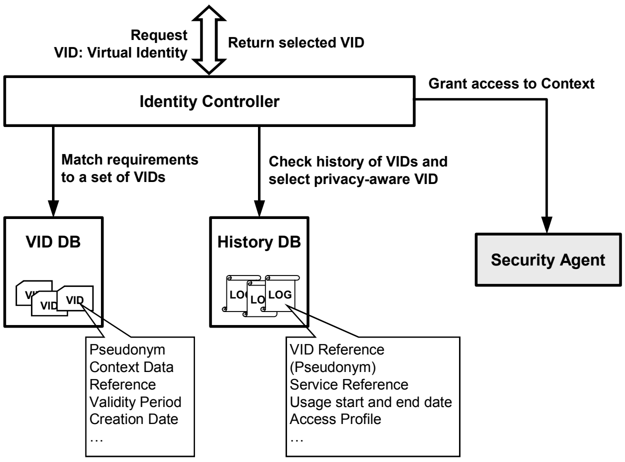

**Figure 2: Architecture of identity management prototype**

The Security Agent shown at the right hand side is not part of the Identity Management, but is a component of the security and privacy management subsystem within Daidalos and a prerequisite to IDM. This component provides basic security functionality, such as cryptographic key generation and management, encryption and decryption, signing of messages/documents as well as creation and verification of (authorisation) credentials. Further, it takes the responsibility of managing access rights in the distributed system, e.g., comprised by different instances of Context Manager.

[1] For the sake of simplicity we assume uniqueness of context information description within every set of personal data composed to one Virtual Identity.

The main control of IDM is located in the component named Identity Controller, which takes requests for Virtual Identities from users – via other Daidalos platform subsystems – and from local (background) services. The Identity Controller then searches the VID database (VID DB) for already existing VIDs that matches the properties stated in the request. Such properties include – besides others – a set of context information descriptions required to provide the service, a name or identifier[2] of the recipient of the VID, the purpose the VID will be used for as well as the VID usage time[3]. These properties will be later on logged into the History DB, which is the memory of IDM and represents the knowledge to prevent personal data aggregation at service instances and thus linkage of different VIDs.

As indicated above, after searching for generally matching VIDs within the VID DB the Identity Controller will then further narrow down the set of appropriate VIDs by means of the History DB – thereby checking which VID contains the least revealed information. This selection process is the major focus to be investigated in the future.

Finally, IDM possesses a reduced set of existing VIDs that match the current request. Currently, the Identity Controller selects an arbitrary VID of this remaining set and passes it to the requestor. In future we will focus on enhanced mechanisms.

In Figure 3, we show a simplified message sequence chart of a user requesting a service (on the right hand side). The service needs to access some context information of the user and thus the privacy policy negotiation needs to provide a suitable Virtual Identity of the user for the service to use. Naturally, there will also be a Virtual Identity of the service for this particular service session associated with the user. The service uses the provided credentials to access only that portion of context, which is associated with the user's VID. In addition, access control information is provided to the user's Context Manager, which acts as the actual Access Control Enforcement point. The following logical sequence describes the Daidalos privacy protection concept in detail:

- User requests a service
- User's pervasive service management initiates the privacy policy negotiation process by requesting an identity for the service usage (after discovering the service which is omitted here)
- User's security & privacy manager carries out the negotiation process with the counter party (service's security & privacy manager)
- User's and service's security & privacy managers compose new or reuse respective Virtual Identities and enclose corresponding credentials
- New (or reused) identities are provided for user's and service's pervasive service management
- Before user's identity is handed over, the access control is updated, providing enough information on rights to access user's (personal) data
- The final step is to confirm (or reject) the service usage and confirm (or reject) the provision of a service
- Retrieval of personal data by a service is checked on the basis of access rules by a user's privacy & security manager

[2] Note, that this might be a VID of the recipient itself.

[3] In cases where the VID usage time is not known in beforehand, the Identity Controller assumes indefinite usage time. This implies that the other Daidalos subsystem is in charge of notifying IDM of the end of usage. This is indicated in Figure 2 by "Return selected VID".

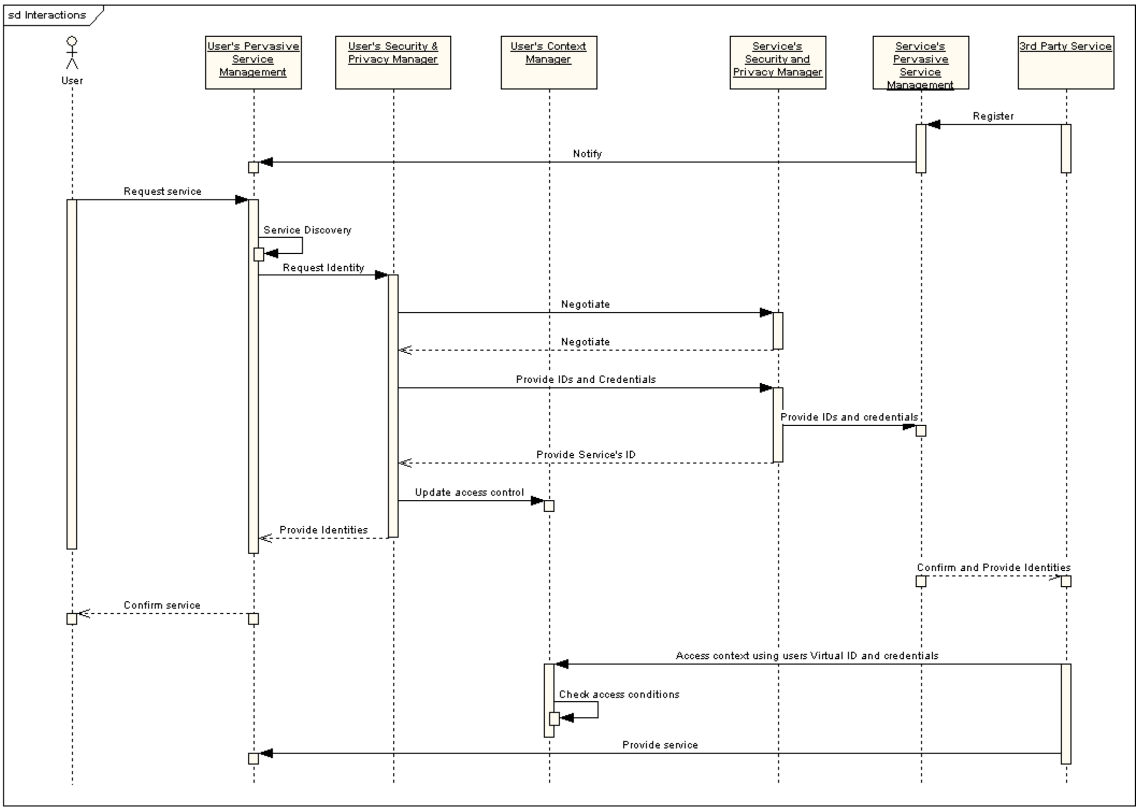
sd Interactions
User
User's Pervasive Service Management
User's Security & Privacy Manager
User's Context Manager
Service's Security and Privacy Manager
Service's Pervasive Service Management
3rd Party Service
Register
Notify
Request service
Service Discovery
Request Identity
Negotiate
Negotiate
Provide IDs and Credentials
Provide IDs and credentials
Provide Service's ID
Update access control
Provide Identities
Confirm and Provide Identities
Confirm service
Access context using users Virtual ID and credentials
Check access conditions
Provide service

**Figure 3: Simple MSC showing the interaction of the user with a service requesting access to the user's context provider**

## 5. Summary and Discussion

A Virtual Identity composes a set of personal data for a specific service. This information is part of the overall user's context, which reflects his current privacy protection based on the VID approach. This approach has a crucial challenge: The VIDs of a user must not be linked by any service, or at least the user – or some entity acting on behalf of the user – must be aware of any linking leading to privacy risks. Moreover, the VID approach has to be supported from the very first stage in the design and onwards. Beneath the communication system, including the authentication and authorization system, the VIDs of a user must not be linked whilst being capable to nevertheless guarantee legal enforcement and charging of pseudonymous users. In Daidalos, the respective subsystems, like A4C, are specifically designed to reflect protection of VIDs against linking, thus supporting the work presented in this paper.

There are also implications on other subsystems of Daidalos. As one example, we discuss the impacts on personalisation in the following. Personalisation is an important aspect of any pervasive system. It plays a role in determining which device to use, which network to access and which software to select and under what circumstances. To achieve this it needs to be guided by user preferences. These take the form of a collection of attributes and rules.

A simple example of such a preference may be one that determines whether or not to redirect incoming communications, and where to redirect them to. This may depend on the time of day/day of the week, on the user's location or on the messages themselves. It is easy to imagine sets of rules that the user would not wish to be seen by his/her employer, spouse, junior staff, etc. Thus privacy and security of such preferences is essential.

These preferences are stored in a user profile. In order to maintain personal privacy, a separate user profile is kept for each VID. This not only ensures protection of the information but also allows the user to have different preferences for different Virtual Identities – thinking of the introducing example of John.

However, this does raise a problem in another area. One of the main difficulties with personalisation is the amount of effort required to build up an adequate set of user preferences. In order to address this, one needs to use some form of learning to identify patterns in the user's behaviour and suggest new preference rules that can be automatically added to the user's profile if the user agrees. However, by compartmentalising the profile into separate VIDs, this makes it more difficult to identify preferences.

Furthermore, if a preference is identified for one VID, it cannot automatically be transferred across to any other VID for the same user. This could be confusing to a user who in general will not keep track of which preferences have been identified for which VIDs. This is an issue, which we will be looking at within Daidalos.

Overall, Daidalos presents a sophisticated system approach that will tie different technologies and innovations together in a seamless way with security, privacy, cost, quality of service and efficiency all factored into the process of pervasive systems. Concerning privacy protection in pervasive environments, it is very important to consider holistic solutions taking into account the application as well as the network, which can be done in Daidalos due to the shared competence of the partners in all those areas.

## 6. Acknowledgement/Disclaimer

The work described in this paper is based on results of IST FP6 Integrated Project DAIDALOS, funded under the European Community's Sixth Framework Programme, whose support is gratefully acknowledged. Apart from this, the European Commission has no responsibility for the content of this paper. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

## 7. References

[1] [D411] Daidalos Consortium: "Daidalos Pervasive Systems Scenarios, Requirements, and Architecture", Deliverable document, June 30th 2004.

[2] M. Morgenstern, "Controlling Logical Inference in Multilevel Database System," Proceedings of the IEEE Symposium on Security and Privacy, 1988.

[3] A. Pfitzmann and M. Köhntopp: “Anonymity, Unobservability, and Pseudonymity – A Proposal for Terminology”, Workshop on Information Hiding, April 2001

[4] 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.

[5] Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications).

[6] M. Winslet: “An Introduction to Trust Negotiation”, Department of Computer Science, University of Illinois, Urbana IL 61801, 2003.

[7] G. Yee and L. Korba: “Semi-Automated Derivation of Personal Privacy Policies”, May 2004.

[8] D. Olmedilla, T. Lara, A. Polleres and H. Lausen: “Trust Negotiation for Semantic Web Services”, January 2004.

[9] K.E. Seamons, M. Winslett, T. Yu, L. Yu and R. Jarvis: “Protecting Privacy During On-line Trust Negotiation”, 2003.

[10] N. Li and J. C. Mitchell: “Role-based Trust-Management Framework”, Proceedings of the DARPA Information Survivability Conference and Exposition, IEEE CS Press, 2003.