# Privacy conscious architecture for personal information transfer from a personal trusted device to an HTTP based service

Pekka Jäppinen, Mika Yrjölä and Jari Porras

***Abstract*— Modern services request personal information from their customers. The personal information is not needed only for identifying the customer but also for customising the service for each customer. In this paper we first analyse the existing approaches for personal information handling and point out their weaknesses. We desribe an architecture for the delivery of personal information from the customer to the HTTP based service in the Internet. For personal information storing our architecture relies on a mobile device, such as a customer's mobile phone. The access of the service is conducted with a traditional desktop computer. The information is transmitted to the service on request via a desktop computer that fetches the information from a mobile device over a wireless link.**

**The goal of our approach is to simplify the use of services by helping the customer to provide the required personal information. Furthermore our approach is designed so that existing services require only minor changes. We introduce methods for the customer to control his own privacy by providing notation to define the required security measures for automated data transfer. Finally we discuss the possible security risks of our architecture.**

***Index Terms*— personal information, usability, Internet service, personal trusted device, privacy, service, bluetooth.**

## I. INTRODUCTION

*John manages to publish a paper in a conference and now he has to arrange his trip to the conference. First he connects to the conference website and registers himself as a conference presenter. In the registration process he types in his personal information e.g. name, address, and phone number. After the conference registration he connects to the airline web page and reserves plane tickets to the conference. Again, he has to type in the same set of personal information. A similar procedure is repeated when John reserves a hotel room and orders the proceedings of the previous conference from the online bookstore.*

The example above presents a typical case where services request information about the users of the service. In addition to personal information also the wishes, preferences or needs of the users are often queried by the web-sites. This gathered information, so called user profile, is then used for identifying the user as well as for personalising the content of the page like in the web based bookstore Amazon.com. According to LaRose et al. [1] , the information collected by web based services typically includes last name, credit card number, demographics, telephone number, street address etc., which are all very private and can be used to identify the user. Similar results are presented by Heikkinen et al. [2] in their survey of over 100 websites offering personalised services. According to their research, most of the websites required personal information similar to the example above. The use of Internet services leads to a situation where the user has several different profiles in the web [3]. Although the information is gathered for personalisation purposes, the user centricity is decreased as the user is confused by what data each of the services holds in their user profiles.

In this paper we base our work on the fact that services request and use personal information and users provide this information to the services. We take a user-centric approach to the personal information management problem as we believe that the user should have ownership of his/her own profile data. We see that the data should be stored locally as the local storage is fundamentally more private. Therefore, in our approach the data containing personal information is stored at the user's personal trusted device, e.g. mobile phone. We present an architecture that uses this local profile storage in HTTP based services.

The rest of the paper is organized as follows. Chapter 2 considers the personal information and the location where it should be stored. This chapter also contains some related work from the field of our research. Chapter 3 describes the general architecture of our mobile device based approach. Chapter 4 discusses the personal information, its handling and the methods used to protect privacy. Chapter 5 describes the communication of different pieces of the architecture and shows how the architecture operates. In chapter 6 we go briefly through the potential attacks against our system. Finally chapter 7 concludes the paper and discusses future research.

## II. PERSONAL INFORMATION MANAGEMENT

The term personal information can be interpreted in several ways. In personal information management (PIM), personal information means information that is owned by the user, e.g. to-do list, calendar notes or phone numbers. The personal information required for the personalisation of services is information about the given person such as name, address and different kinds of preferences. The latter is what we consider personal information in this paper. In [4] personal information is shortly analysed and then divided into user profile and identities according to the view point. If we are looking at the personal information from the user-centric point of view

Manuscript received June 29, 2004; revised May 09, 2005, and November 09, 2005. This paper was presented in part at the Conference on Software, Telecommunications and Computer Networks (SoftCOM) 2004.

Authors are with Lappeenranta University of Technology, Department of Information Technology (e-mail: {Pekka.Jappinen, Mika.Yrjola, Jari.Porras} @lut.fi)

then we should talk about identities rather than user profiles collected by the services. Although the user-centricity is the main goal of our research, we use the general term personal information in this paper.

In our research the more important issue, than exact content of the personal information, is the management of the data. We have taken the existing approaches as the base for our work and propose an approach that combines the best parts of each approach. Figure 1 presents all these approaches, namely 1) registration, 2) database at network, 3) browser and 4) PTD based approaches. The existing approaches 1) - 3) are shortly introduced in this chapter and motivation for our approach is given.

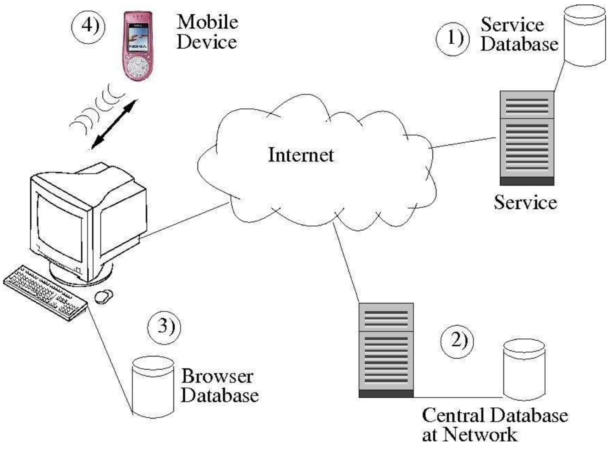


Fig. 1. Different locations for personal information

Many of the current Internet services utilize the registration approach. Users have to register for the service before they can start using it. The service stores the information about the customers, i.e. user profile, into its database for further usage. Thus next time the user accesses the same service he doesn't have to provide the same information again. Instead, the service looks up the user profile from its database. In order to find the correct personal information from the database a user authentication is required. Besides the basic identifying information, services may request the user's preferences, follow the user's behaviour on the server and store the gathered information into an appropriate database. The stored information is then used to personalise the service i.e. adapt the service outlook and content into a more customer friendly form. Effective personalisation will provide better customer satisfaction and thus increase the customer's loyalty towards the service [5], [6]. Effective personalisation also requires a lot more information about the customer than just simple contact data.

However, the service database is not always the most optimal place for the information as claimed in [7]. As the variety of the used services grows, the places to where the user has to provide his personal information grow and the confusion about the contents of the user profiles increases. If the information needs to be changed the user needs to know all the places where the information is stored. It is unlikely that the user remembers all these places thus resulting in contradictory information. Another problem arises when considering the security of the personal information. When the information is stored in the service database, the user has to trust that the service provider is capable of storing the information securely enough. The more services the user uses, the bigger is the chance that one of them has weaknesses in their security system. To minimise the amount of databases the personal information is stored in, the customer can easily decide to concentrate on to only few service providers.

A second possibility is to store the personal information in a single database on the network. This database can be administrated by a trusted third party or the customer. Some of the single sign-on (SSO) approaches such as Microsoft .NET passport [8] and Liberty Alliance [9] can be used to provide a third party based solution for the personal information storage problem. However, originally the .NET passport required the service provider to pay a fee to Microsoft. This meant that small service providers could not afford to adopt the passport. It is also unlikely that temporary services like conference registrations are willing to do the extra work required to join in the SSO system especially when all they need is simple registration information for a one time event.

In Liberty architecture, service and identity providers form circles of trust in which the participants transfer information about the user. The creation of a circle of trust requires negotiations between the participating partners. Like in the .NET passport this is unsuitable for one time services provided by small service providers. In these approaches the user has to trust the third party and that the third party uses secure methods when handling the user's sensitive data. Also the trusted party has to be the same party for both user and service provider. If there are several third parties providing similar personal profiles, the service provider has to have a contract with most of them, in order to not exclude any users. Obviously, information stored on third party servers, is available only when a connection can be formed between the service provider and the servers. Thus the use of the service relies not only on the reliability of the network between the customer and service but also the network between service and the third party. This problem is graver in countries that have poor international Internet connections. Finally, the third party service may change the way it functions and the terms of use. For example Microsoft has stated that they will cease to offer passport functionality outside their own services. In Microsoft and Liberty approaches the user has no control over his/her profile, which is a serious drawback.

Other centralized approaches have been proposed. Koch proposes IDRepository that allows users to own their own profiles [10]. At the same time this approach supports complex user profile attributes. In the Wireless World Research Forums (WWRF) book of visions Bettstetter et al. suggested a user controlled personal profile server [11]. A more detailed solution was described by Thai et al. [12]. Their Integrated Personal Mobility Architecture relied on the customer's home network as a location to store personal information. Compared to third party approaches the user controlled database is easier for service providers since they don't have to have contracts with different third parties. The data is available for services when the profile server is up and running. A security conscious user will turn the service on only when needed, thus increasing security.

The third obvious location for the personal information storage is at the user end. New web browsers have the capability to store some information for the user. Thus, the information is stored in the place, from where it is easiest to use and update. There exist two approaches that can be used to determine what information will be filled in an empty form. In the first approach the stored information depends on the web page. The browser remembers what the user has typed into the form fields, when the form was filled in before. In the second approach the web page supports the browser's personal information scheme, so that the browser itself can determine what information is given in what field [13]. Although the web browsers will enhance this personal information storage functionality rapidly, it does not help the mobile or nomadic users. Mobile and nomadic users typically access the services from several different terminals at different places.

All the aforementioned approaches have problems in information access, security and management issues. Although offering personalised services whenever accessible, the service database approach suffers from security and management issues. The biggest challenge is in keeping the information coherent on different services. The trusted third party approach offers coherence but suffers from the access issue. It is possible that the user may connect to the service but the trusted third party is non-accessible. Accessibility is further improved by the use of a browser based approach. With this approach the personal information is accessible and updatable if the same browser or computer is used all the time. Unfortunately, this type of local information approach does not support nomadic users. To overcome these problems we have developed an approach where personal information is stored in the customer's mobile device. The personal information can be requested from there by the service either directly or via the service accessing device e.g. the desktop computer. In this approach the personal data is under the customer's control i.e. the customer always can access the data and can define who is allowed to fetch the data automatically. The data is available for the service when the customer is using the service. Therefore the service does not have to store all the data in its database.

Similar approaches have been proposed by many researchers. The EU funded Simplicity project [14] concentrates on the use of services through different terminals and devices. In order to provide seamless usage of services in different devices the Simplicity project proposes the use of a so called Simplicity device to store the user profile. Although several possibilities for the storage are given they conclude that the SIM + smart phone combination is not very far from the ideal storage and processing device. Thus from this point of view the Simplicity project is quite close to our approach. Riche et al. proposes in [3] a distributed client side approach where the user profiles are stored in several user devices. Each device can store a part of the profile. Although the approach is very intuitive as the user often uses a limited number of devices, the challenge is in the synchronization of the profile data, as seen in the paper. This concept is out of the scope of this paper as we consider only a single device as the storage for personal information.

## III. PTD BASED APPROACH

In this paper we present an approach that supports nomadic users and offers manageable, accessible and private storage of personal information. In our approach the personal information is stored in a single personal mobile device, i.e. Personal Trusted Device (PTD). The Personal trusted device term is used in many places. In electronic commerce/banking the personal trusted device means mainly a device that offers strong authentication of the user for the services [15]. Although authentication is important in some cases, it is not the main issue in our approach. Security issues of the PTD are also emphasized in the presentations of [16], [17]. In [17] the PTD is defined with the more extensive scope in mind, but the discussion is mainly on the security and privacy level. In our approach the definition and the purpose of PTD is more on the use of services through the PTD than on the security issues of the communication. Security issues are important, but they are only a part of the bigger scope of the PTD approach. For us personal means that the device is used only by one person: the owner of the device. This means that that person can trust that the device holds the right information and it is not tampered. The device has its own authentication method to verify that the owner of the device is handling it. PTD may be used for authentication purposes but it can also be used to enable anonymous personalisation of the service [18]. For us, PTD is a device that is carried around almost at any time and any place and it is used in local (personal area networking, local area networking) and in global communications. PTD is the centre or the hub of all personal communication the user wishes to perform. Our concept of the personal trusted device in personal communications is further explained in [19].

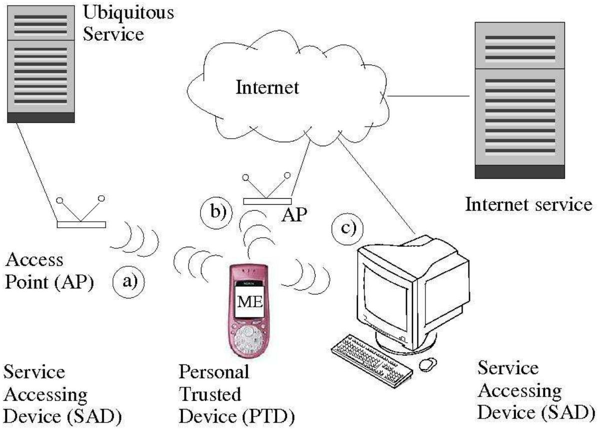

Ubiquitous Service
Internet
Internet service
b)
AP
c)
Access Point (AP)
a)
ME
Service Accessing Device (SAD)
Personal Trusted Device (PTD)
Service Accessing Device (SAD)

Fig. 2. PTD and ME in service access

PTD can be used in several different situations. Figure 2 presents example cases where PTD can be used as a part of the service access. In a) and b) the services are used through the PTD whereas in c) PTD is just a complementary part of the service access. The actual access to the service in c) is conducted via a fixed terminal such as a desktop computer. In this paper we concentrate on case c). Approach a) is presented in details in [18].

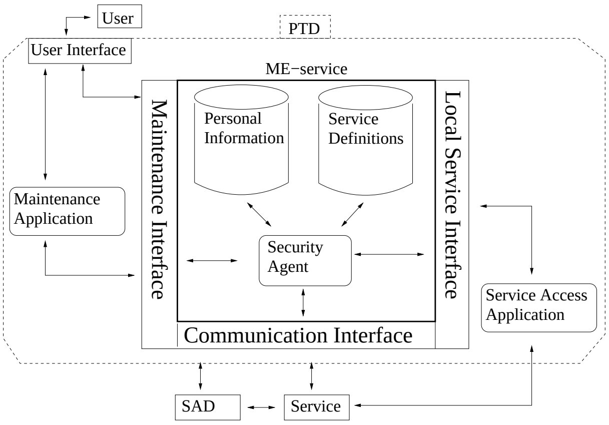


Fig. 3. Architecture of ME - service

The concept of PTD can be further expanded by considering PTD as personal information storage, as was done in [20]. The idea of PTD, e.g. mobile phones, is that it stays with the user all the time and thus the information stored in it can be accessed and used whenever necessary. In [20] a separate service, i.e. mobile electronic personality - ME, for personal information handling was defined. ME itself is a part of PTD as presented in Figure 2.

The ME service (presented in Figure 3), consists of two databases and a security agent. All the personal information is stored in the personal information database while information about the services and their authentication credentials are stored in the service definitions database. The security agent handles the requests of personal information. Before any piece of data is sent, there is a check to determine whether the requester has rights to the information [20].

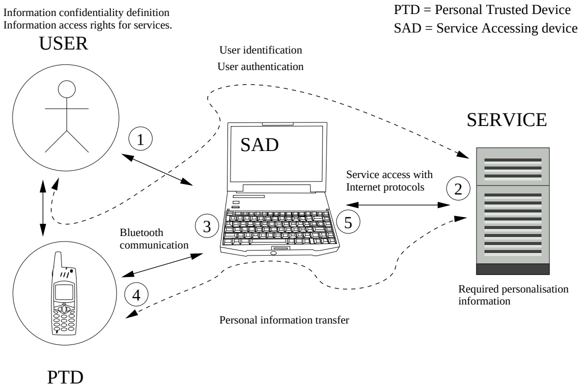

Information confidentiality definition
Information access rights for services.
USER
PTD = Personal Trusted Device
SAD = Service Accessing device
User identification
User authentication
SERVICE
SAD
1
Service access with Internet protocols
2
Bluetooth communication
3
5
4
Required personalisation information
Personal information transfer
PTD

Fig. 4. General service access architecture and operations

The goal of this paper is to present how the personal information stored at the PTD can be used in traditional HTTP based services. The architecture for personal information transfer from a personal trusted device to a HTTP based service consists of four physical parts (Figure 4) User, personal trusted device (PTD), service accessing device (SAD) and the service. The service itself can be any type of service provided in the Internet. SAD is used by the user to connect to the service and is usually a desktop or a laptop computer. The PTD that contains the personal information is owned by the user and in general it can be a PDA or a mobile phone.

Figure 4 shows the basic operations and parts of the system. The general process of service usage where personal information is accessed from PTD can be divided into 5 phases.

1) The user connects to the Internet service with the SAD by using generic Internet protocols such as HTTP.
2) The service requests personalization information from the SAD.
3) SAD connects to the PTD.
4) PTD transfers the requested information to the SAD.
5) SAD forwards the information to the service.

Depending on the type of information that is transferred, user action might be required between steps three and four. User action ensures that sensitive information, such as credit card numbers, is not transmitted without the user's consent. The privacy aspects have to be taken into account before the actual communication procedures are defined in more detail. If the user identity is required by the service, PTD may be used as an authentication token. The authentication credentials can be transmitted through the same route as personal information.

## IV. Personal Information and Privacy

From a privacy point of view, different pieces of personal information vary from each other. There exists identifying information such as a real name or address that reveals the person's identity and thus may provide risk towards the privacy. Then there is the general preference information such as preferred type of sports, which alone does not reveal the identity, but can be used for service personalisation. The difference in the sensitiveness of the data should affect also on the handling of such data.

In the PTD based approach the data is stored at the user's PTD, from where it is fetched by the SAD when required. From a usability point of view, the fetching of data should happen automatically without extra effort from the user. On the other hand sensitive data should not be transmitted without the user's consent. The decision what information is transmitted automatically from PTD to SAD and which information requires explicit approval, has to be done by the user. In our model each piece of personal data has a security level attribute, which is a value between 0 and 9. The bigger the value the more sensitive is the data. This value is set by the user when the data is added on the database of ME. This approach was chosen to keep the system simple and understandable for the user although hardcore network users may prefer more options on configuration.

For each security level, the user can define what kind of security measures have to be taken into account, before the transaction of personal information can be committed. For ME, three distinct security measures have been defined:

1) Service authentication
2) Encryption
3) User interaction

Service authentication means that before the personal information is given, the service has to be authenticated as a trusted service. From the communication architecture point of view, three different grades of authentication can be identified. On grade one, service authentication is not required and the information is delivered to whoever requests it. On grade two, the service authentication is done by the SAD which informs the PTD about the service authentication. This approach is useful when SAD is a trusted device, such as the user's personal computer, that is used frequently to access services. Therefore, the SAD can be trusted to perform the service authentication. When dealing with untrusted SAD, such as a public computer, service authentication should be done by the PTD. This grade three authentication requires that a service certificate is stored at the PTD. Modern mobile phones and PDA's have enough memory to store the certificates and have enough computation power to accomplish the authentication in adequate time. In order to determine whether the used SAD is actually trusted SAD, the PTD should hold certificates of the trusted SADs. When the communication between SAD and PTD is created authentication of the SAD should be executed.

From an encryption point of view, the transmission of personal information can be either considered to be between PTD and the service or divided into two halves, namely SAD-PTD and SAD-service communication links. Encryption between the SAD and the service protects information eavesdropping on Internet communications, while SAD-PTD communication protects against local eavesdroppers. The encryption decision for these links can be independent from each other. Often the used service is accessed by using secure HTTP, where all the transmitted data is encrypted. When requesting the personal information, SAD should inform the PTD about the used encryption algorithm and the key size. This information provides the base for the PTD to decide whether the encryption is adequate for data to be sent. For SAD-PTD communication, wireless communication standards e.g. Bluetooth, have encryption mechanisms specified in them. ME can then decide whether the included encryption method is secure enough. When the encryption between separate links is not enough, encryption has to be implemented between the PTD and the service. Such an approach also prevents the SAD from eavesdroping personal information and is therefore recommended when the SAD cannot be trusted, e.g. when the service is accessed from a public computer at the library. Encryption between SAD and PTD requires encryption key exchange that will add some overhead to the actual communication and some complexity to the communication model.

User interaction means that the user explicitly expresses, by pressing the appropriate buttons on PTD, whether the given piece of information can be sent to the service or if transaction should be cancelled. In order to keep the service personalization as transparent as possible, the user interactions should be minimized. Therefore, only the most important information at highest security level, such as the credit card number, should require explicit user confirmation. Since the user might trust more on some services than others, the requirement for the explicit user action can be defined for each service separately. Just like the case with the personal information, also the services have appropriate security levels attached. The personal information, that has a lower or equal level than the level of service, may be sent without the user's explicit approval.

The used security methods are decided by the PTD and are based on the definitions made by the user at the configuration stage. For example, the user might require that for level 4 information the service has to be authenticated and the transmission should have at least a 128-bit encryption, while for level 5, a 256-bit encryption is required.

A lot of the privacy related research concentrates on defining the level of the user identity. In his PhD thesis Goldberg called these levels as nymities starting from verinymity where the user's real identity is known to real anonymity where nothing is known about the user's identity [21]. Several research projects such as DAIDALOS [22], Mobilife [23] and PRIME [24] have defined ways of handling a variety of pseudonyms the user may have when using different types of services. In our approach the user identity and nymity is not considered. User authentication, whether with pseudonym or real identity, is not conducted by ME. On the other hand, the user preference information can be acquired by the service even if the user has not been registered on the service. Thus it is possible for the user to remain anonymous and still gain the benefit from personalisation. For example, the user may get recommendations about the book types he prefers at the online bookstore without logging in. Once he finds something interesting he may log in to the service and reveal his identity to order the book. The user may not have even used the given service beforehand and gain the personalised recommendations anyway.

## V. SERVICE USE AND COMMUNICATION

Let's go back to the example in the beginning of the paper where John was registering to the conference, but now he uses our architecture. Figure 5 describes the messages and actions required in such use. First John connects to the conference webpage just like before: he types in the conference URL and web browser then sends the request to the service. The conference server provides a web page for registration, which also contains a JavaScript that starts a plug-in program on the browser. The plug-in program handles the communication between the SAD and the PTD. It first forms the connection to the PTD. If there are several devices providing the personal information, John has to select his own PTD from a list. Then the plug-in requests the required personal information from PTD. Depending on the security settings John may have to accept the transmission explicitly from his PTD. The plug-in fills in the registration form with the acquired data. Finally the data is sent to the service when John presses the *send registration* button on the web page.

In order to determine, which part of the transmitted data is the name and which part is the address, a notation for the data is required. Unfortunately, personal information is marked in various ways depending on the situation. Service providers have their own proprietary ways to store it for their own use as do web browsers [13]. E-commerce has its own model for

```
<xs:simpleType name="MESecLev">
     <xs:restriction base="xs:integer">
         <xs:minInclusive value="0"/>
         <xs:maxInclusive value="9"/>
     </xs:restriction>
 </xs:simpleType>
 <xs:element name="FullName">
     <xs:complexType>
         <xs:simpleContent>
             <xs:extension base="xs:string">
                 <xs:attribute name="SecurityLevel" type="MESecLev"/>
             </xs:extension>
         </xs:simpleContent>
     </xs:complexType>
 </xs:element>
```

Schema 1: Part of the schema for personal information

field names that is designed for e-commerce purposes [25]. For contact information a vCard standard has been defined [26], [27]. In addition, several independent projects have defined XML based markup for the name and the address information [28]. Unfortunately, none of these are widely adopted in services so far. Also the aforementioned notations do not take into account the privacy issues stated in previous chapter nor do they include preference information.

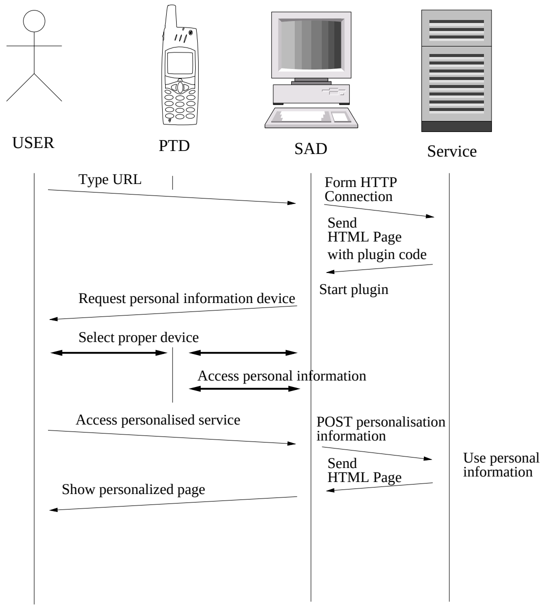

USER
PTD
SAD
Service
Type URL
Form HTTP Connection
Send HTML Page with plugin code
Start plugin
Request personal information device
Select proper device
Access personal information
Access personalised service
POST personalisation information
Send HTML Page
Use personal information
Show personalized page

Fig. 5. Generic HTTP-based service access

To fulfill the needs of our approach we have developed a simple XML schema for personal information (Schema 1). The schema contains tag names for different personal information pieces and a MESecLevel attribute that states the security level of the given piece of information. The information is stored in the PTD following the XML schema. The same tag names are also used by the service when it requests personal information. Thus when a request arrives, ME simply browses through the DOM (Document Object Model) tree of the personal information database to find a tag that is equal to the one in the request.

To support our architecture, the conference organizer has to make some changes in the registration webpage. First of all there has to be a JavaScript code that starts the plug-in. Then the tag attributes that state which field is for surname and which is for address should follow the naming scheme of ME. Although a lot of research is going on in the semantic web research field for matching different notations to each other using ontology languages, the current version of the plug-in does not have such a possibility.

From the example above, it can be seen that the communication in our architecture can be separated in two separate parts: Service-SAD and SAD-PTD. Since the service is accessed via an HTTP protocol, the communication between the service and SAD naturally relies on the HTTP too. The communication between SAD and PTD is independent of the service access protocol. Therefore, even if the service access method changes from HTTP to another protocol, the defined SAD-PTD communication protocol remains the same. On SAD, the SAD-PTD communication is handled by a simple program such as a web browser plug-in [29]. This enables the possibility to use the method defined for personal information transfer between the mobile device and the transparent service provided through an ubiquitous network [30].

In our implementation Bluetooth wireless technology [31] is used for the SAD-PTD communication. Bluetooth consumes little power so that it can be used in mobile devices that usually rely on battery power [32]. In addition to that, the Bluetooth support on mobile devices is getting more common. Therefore, it is very likely that in the future most users that have a mobile device have also the Bluetooth communication capability. Bluetooth supports a variety of protocols and their use in different cases is defined in several profiles [33]. For our implementation an OBEX protocol was selected due to its simplicity. OBEX is also supported by the IrDA standard [34]. Therefore, the OBEX protocol stack is already implemented in most of mobile devices that support infrared communications.

### A. Service-SAD communication

Communication between the service and the SAD is done by using an HTTP protocol. The messages needed for the personal information access, are encapsulated in the payload of the HTTP messages. This requires additional functionality to the SAD as well as the service, but allows the end user to use already familiar ways to access the services.

Since plug-in is used for handling the communication between SAD and PTD, it has to be activated by the HTML document that the SAD requested from the service. The activation is done by using JavaScript on the web page. The actual request can be either part of the HTML document or a parameter on the JavaScript that runs the plug-in.

In the HTML document approach the request is encoded on the web page throuhg a distinguished field where ID is a PERSONALISATION REQUEST. Plug-in can find this field by browsing the DOM tree of the document. The field contains a list of personalization information that the service requests. If the HTML Document holds a form to be filled, the field names on the form can be notated following the XML-schema for the personalization information. This way the plug-in can check the requested information from the field names. If the request is delivered as a parameter the request is decoded right at the SAD by the plug-in program. This is faster than browsing through the DOM tree. Support for both ways for request, helps the service providers to adapt the service to apply the ME approach.

No matter which way the request is done, it consists of a list of wanted personal information and optionally the certificate of the service. The requested personal information field names should match those that are defined in the XML schema for personal information. These are then forwarded to the PTD.

The response to the service is encapsulated also in the payload of the existing protocol message, such as HTTP POST. The response may contain the requested information as an whole, special code for providing reason for failure not getting information or just part of the information and explanation code. Again the personal information is tagged in XML following the XML schema for the personal information and is part of the response document.

### B. SAD-PTD communication

The communication between the SAD and the PTD is done in two separate steps. The first step is to discover the Bluetooth devices that support ME (Figure 6). After discovery, the retrieval of the personal information from PTD can be done (Figure 7).

In first step, all the devices in the range of the SAD are called via service discovery protocol (SDP) to find out devices that support the ME service and therefore have the capability to transmit personal information. SAD then shows all the found devices on its screen, from where the user can select his own PTD. PTD's Bluetooth hardware address (BDADDR) can also be stored in the user's plug-in configuration file to avoid the slow discovery process. After the correct Bluetooth device is determined, the normal Bluetooth connection forming takes place. When the OBEX connection is formed between the SAD and the PTD, the PTD acts as an OBEX server and SAD acts as an OBEX client. Now the SAD can request personal information from the PTD.

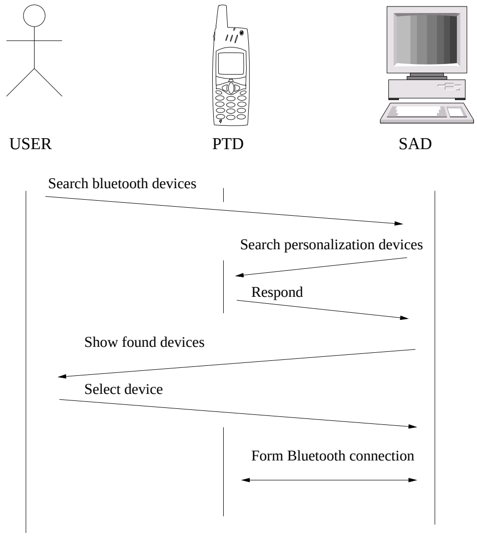


Fig. 6. Forming SAD-PTD Bluetooth connection

Communication between the SAD and the PTD should support the privacy options defined in section 3. Thus SAD has to provide the necessary information about the used service as well as the security status of the SAD-service connection. The communication should be done with existing protocols such as OBEX. The request and reply are then encapsulated on the protocols data field.

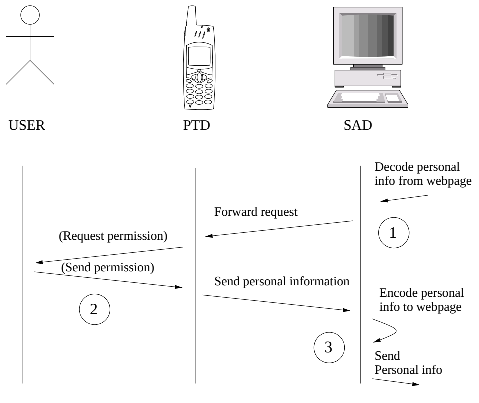

USER
PTD
SAD
Decode personal info from webpage
Forward request
1
(Request permission)
(Send permission)
2
Send personal information
Encode personal info to webpage
3
Send Personal info

Fig. 7. Personal information retrieval from PTD by SAD

The personal information request consists of five fields: size of service public key, service public key, authentication level, encryption level and requested personal information fields. These fields are separated from each other by a colon. If SSL (Secure Sockets layer) communication is used the plug-in can get the service's public key from the browser's database, otherwise the key should be in the incoming web page. If there is no public key available the size of the service public key is 0 and the key file is left empty. The authentication level defines whether the service is authenticated by the SAD or not. Encryption level defines the size of the encryption key used for the symmetric encryption of the SAD-service connection. If no encryption is used the encryption level is set to 0. In practice this means that when SSL is used for service access, authentication is set to 1 and the encryption level is bigger than zero. When using OBEX the request is encoded in the OBEX GET message.

For the personal information request there are two types of responses: personalization response, which contains the requested personal information and error response, which tells the reason why the requested information was not delivered. The reply is encoded as one object, which is sent to the SAD in the OBEX response message.

## VI. Security Risks

The targets for possible attackers in the architecture can be divided into two groups: communication channels and hardware. For communication between the SAD and the service our architecture does not provide any real additional security measures. PTD may conduct the authentication of the service using the standard X.509 certificates and asymmetric cryptography on behalf of the SAD. This approach is similar to commonly used SSL connections

The connection between SAD and PTD can also be compromised. There are several attacks developed against Bluetooth connections that utilise the insecure implementation of the Bluetooth stack on host devices [35], [36]. The short communication range of Bluetooth is not a valid security point either as there has been developed a device called the bluesniper rifle that can intercept a Bluetooth signal over kilometers of distance [37]. Even though Bluetooth uses a strong encryption algorithm it is possible to force connected devices to conduct a new key exchange process and after that eavesdrop on the communication[38]. Therefore, the connection has to be secured at the application layer.

The hardware targets for attacker may be the PTD, service or SAD. First of all the attacker may steal PTD and then take the information out from it directly. To prevent direct data theft from the device the user has to be authenticated before the preferences data on the database can be read or edited. An attacker may also try to pretend to be a trusted SAD or service in order to acquire information from the device. Used authentication methods based on asymmetric cryptography are reliable as long as the public key transfer is conducted safely. Man-in-the-middle attack during the keyexchange is a valid thread. Using a public SAD for eavesdropping the personal data transfer between the PTD and service can be prevented by end-to-end security measures stated in Chapter 3.

Although these risks are real our architecture has also some benefits from a security point of view. Attacking against the PTD or the communication channel will only provide the attacker information about one person. A service provider, holding big databases about its customers, is a much more attractive target to the criminal mind. Thus against an attacker who wants to steal identities of hundreds of persons, our architecture can be said to be more secure than existing systems. On the other hand when the target is a single person the advantage is not that clear. The more services the person uses the bigger is the risk that one of them has a vulnerability in their security system through which the personal data can be stolen. Thus it can be stated that the more services used the bigger the benefit, from a security point of view, for architecture.

The current security measures provide basic security, but the existing problems should be further researched. In that research it should be kept in mind that the main goal of the architecture is to add usability. Therefore the security methods should be kept as transparent to the user as possible.

## VII. Conclusion

In this paper we have presented an architecture for transferring personal information stored in a mobile device to the Internet service. The mobile device as personal information storage location provides advantages for information security. Sensitive information is stored only at one place instead of many places, which means that up keeping the information is simple. The mobile device holds the information of only one person, which makes the device less of an attractive target for information thieves.

The information is available for any Internet service with the user permission regardless of user registration. Thus the usability of the services is enhanced as the users do not have to type in repetitive information in every service. The approach also enables the possibility to provide an anonymous user a personalised view of the service. The used approach requires little changes to the service itself as most of the work is done by an external downloadable browser plug-in. Therefore, the described approach can be applied in several existing services with little cost.

Further research is required to define an effective user interface for controlling personal information. User interface research should answer the question on how the user can easily define, which information can be acquired automatically by services and which services are actually allowed to do so. Additional research should be conducted on the security issues of the data transfer. Furthermore, support for identity management and pseudonyms should be researched.

## References

[1] R. LaRose and N. Rifon, "Your Privacy is Assured – Of Being Invaded: Web sites with and without privacy seals," in *Proceedings of the IADIS international conference e-society 2003*, pp. 63–71, June 2003.

[2] K. Heikkinen, J. Eerola, P. Jäppinen, and J. Porras, "Personalized view of Personal Information," *WSEAS Transactions on Information Science and Applications*, vol. 1, pp. 1050–1055, Oct. 2004.

[3] S. Riche, G. Brebner, and M. Gittler, "Client-side Profile Storage," in *Web Engineering and Peer-to-Peer Computing NETWORKING 2002 Workshops*, LNCS 2376, pp. 127–133, Springer-verlag, May 2002.

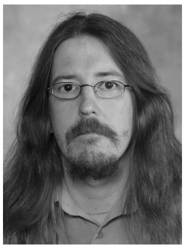
[4] M. Koch and W. Wörndl, "Community Support and Identity Management," in *Proceedings of the Seventh European Conference on Computer Supported Cooperative Work*, (Bonn, Germany), pp. 319–338, Kluwer academic Publisher, Sept. 2001.

[5] L. Ardissono and A. Goy, "Tailoring the Interaction with Users in Web Store," *User Modeling and User-Adapted Interaction*, vol. 10, no. 4, pp. 251 – 303, 2000.

[6] B. Kasanoff, *Making It Personal.* Perseus Publishing, 2001.

[7] P. Jäppinen and J. Porras, "Analyzing the Attributes of Personalization Information Affecting Storage Location," in *Proceedings on IADIS International Conference on E-Society*, pp. 48–55, 2003.

[8] Microsoft, ".NET Passport Review Guide." Available at: http://www.microsoft.com/net/services/passport/review_guide.asp, 7 Oct. 2003. Accessed January 20, 2006.
[9] Liberty Alliance, "Liberty Architecture overview-v1.1." Available at: http://www.projectliberty.org/specs/liberty-architecture-overview-v1.1.pdf, 15 Jan. 2003. Accessed January 12, 2006.
[10] M. Koch, "Global Identity Management to Boost personalization," in *Proceedings of 9th Research Symposium on Emerging Electronic Markets*, (Basel, Switzerland), pp. 137–147, Sept. 2002.
[11] C. Bettstetter, W. Kellerer, and J. Eberspächer, *Book of Visions 2000*, ch. Personal Profile Mobility for Ubiquitous service Usage, pp. 67–69. Wireless Strategic Initiative, 2000.
[12] B. Thai, R. Wan, A. Seneviratne, and T. Rakotoarivelo, "Integrated Personal Mobility Architecture: A Complete Personal Mobility Solution," *Mobile Networks and Applications*, vol. 8, pp. 27–36, Feb. 2003.
[13] G. W. Bauer, "User data management aka Privacy/Convenience Feature." Available at: http://www.mozilla.org/projects/ui/ communicator/browser/wallet/, 26 Jan. 2006. Accessed January 27, 2006.
[14] N. Blefari Melazzi, G. Bianchi, G. Ceneri, G. Cortese, F. Davide, N. Davies, N. Dellas, E. Fischer, T. Frantti, Friday. Adrian, J. Hamarid, M. Helbing, S. Kapellaki, K. Kawamura, W. Kellerer, L. Kotsoloukas, C. Meyer, C. Niedermeier, C. Noda, J. Papanis, C. Petrioli, E. Rukzio, S. Salsano, R. Seidl, O. Storz, J. Urban, I. s. Venieris, and R. Walker, *The Simplicity Project: Managing Complexity in a Diverse ICT World*, ch. 10, pp. 179–211. IOS Press, 2004.
[15] MeT, "Mobile electronic Transactions, PTD Definition Version 2.0." Available at: http://www.mobiletransaction.org/pdf/R200/ specifications/MeT_PTDdef_v200.pdf, 2002. Accessed January 30, 2006.
[16] E. Weippl and W. Essmayer, "Personal Trusted Devices for Web services: Revisiting Multilevel Security," *Mobile networks and Applications*, vol. 8, pp. 151–157, Apr. 2003.
[17] J. Veijalainen, M. A. Haq, and M. Matsumoto, "Privacy and Security Considerations for Personal Trusted Devices." Presentation at the Fifth WIM Meeting. Available at: http://fag.grm.hia.no/fagstoff/vladimao/, 14 Aug. 2003.
[18] P. Jäppinen and J. Porras, "ME - approach for ubiquitous personalization," in *Eurescom Summit 2005: Ubiquitous Services and Applications*, pp. 87–93, Apr. 2005.
[19] J. Porras, P. Jäppinen, P. Hiirsalmi, A. Hämäläinen, S. Saalasti, R. Koponen, and S. Keski-Jaskari, "Personal Trusted Device in Personal Communications," in *Proceedings of 1st International Symposium on Wireless Communication Systems*, (Mauritius), Sept. 2004.
[20] P. Jäppinen, *ME - Mobile Electronic Personality*. PhD thesis, Lappeenranta University of Technology, 2004.
[21] I. A. Goldberg, *A Pseudonymous Communications Infrastructure for the Internet*. PhD thesis, University of California at Berkeley, 2000.
[22] Daidalos, "Daidalos project webpages." Available at: http://www.ist-daidalos.org/. Accessed November 7, 2005.
[23] Mobilife, "Mobilife project webpages." Available at: https://www.ist-mobilife.org/. Accessed November 7, 2005.
[24] PRIME consortium, "PRIME project webpages." Available at: http://www.prime-project.eu.org/. Accessed October 20, 2005.
[25] D. Eastlake and T. Goldstein, "ECML v1.1: Field Specifications for E-Commerce." IETF Standard RFC 3106, Apr. 2001.
[26] T. Howes, M. Smith, and F. Dawson, "A MIME Content-Type for Directory Information." IETF Standard RFC 2425, Sept. 1998.
[27] F. Dawson and T. Howes, "vCard MIME Directory Profile." IETF Standard RFC 2426, Sept. 1998.
[28] Oasis Consortium, "Markup languages for Names and Addresses." Available at: http://xml.coverpages.org/ namesAndAddresses.html, 31 Jan. 2004. Accessed January 27, 2006.
[29] M. Yrjölä, P. Jäppinen, and J. Porras, "Personal information transfer from a mobile device to web page," in *Proceedings of the IADIS International Conference on WWW/Internet*, pp. 485–492, Nov. 2003.
[30] P. Jäppinen and J. Porras, "Transfer of Personalisation Information from Mobile Device to Transparent Service," in *International Conference on Computer Science and Technology*, IASTED, May 2003.
[31] Bluetooth SIG, "Bluetooth Specification 2.0 + EDR." Available at: https://www.bluetooth.org/spec/, Nov. 2004. Accessed January 26, 2006.
[32] J. Bray and C. Sturman, *Bluetooth: Connection Without Cables*. Prentice Hall PTR, 1 ed., 2001.
[33] D. A. Gratton, *Bluetooth Profiles*. Prentice Hall PTR, 2003.
[34] Infrared Data Association, "IrDA specifications." Available at: http://www.irda.org/. Accessed January 26, 2006.
[35] C. Humphrey, "How To: Building a BlueSniper Rifle." Available at: http://www.tomsnetworking.com/Sections-article106.php, 3 Aug. 2005. Accessed November 4, 2005.
[36] C. Gehrmann, J. Persson, and B. Smeets, *Bluetooth Security*. Artech House Publishers, 1 June 2004.
[37] trifinite.org, "Bluetooth vulnerabilities." Available at: http://trifinite.org/trifinite_stuff.html. Accessed November 4, 2005.
[38] Y. Shaked and A. Wool, "Cracking the Bluetooth PIN," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys 2005)*, (Seattle, Washington, USA), pp. 39–50, ACM Press, 2005.



**Pekka Jäppinen** received his MSc and DSc degrees in information technology from Lappeenranta University of Technology in 2001 and 2004. He has been working at the Communications Engineering laboratory since 1995. His research interests include short-range wireless communication, communication protocols, personal information management, security and privacy.

**Mika Yrjölä** studied at Lappeenranta University of Technology from 1995 to 2004. During his studies, he also worked at the communications engineering laboratory from 2000 to 2004. He received his Master's Thesis from the department of information technology in 2004. After his graduation, he has been working at Movial in projects mostly involving embedded Linux. His other interests include amateur astronomy and photography.

**Jari Porras** received his Master of Science degree at the Michigan Technological University in 1993 and the Doctor of Technology at Lappeenranta University of Technology in 1998. He is currently working as a Professor at Lappeenranta University of Technology in the Communications Engineering laboratory. His interests include wireless networks, ad hoc networking, peer-to-peer computing, aspects of grid computing and distributed computing/simulation.