Usable Presentation of Secure Pseudonyms

Katrin Borcea-Pfitzmann
Dresden University of Technology
Department of Computer Science
Dresden, Germany
kf2@inf.tu-dresden.de

Elke Franz
Dresden University of Technology
Department of Computer Science
Dresden, Germany
ef1@inf.tu-dresden.de

Andreas Pfitzmann
Dresden University of Technology
Department of Computer Science
Dresden, Germany
pfitza@inf.tu-dresden.de

ABSTRACT

Privacy-Enhancing Identity Management (PIM) enables users to control which personal information they provide to their communication partner(s) by partitioning their personal information into partial identities for themselves. Since partial identities must not be linkable, they cannot share a global name. Therefore, pseudonyms are used as identifiers.

We discuss in this paper that besides the frequency of their use also the (re)presentation of pseudonyms influences the achievable privacy. Particularly, we point out that conflicting requirements on privacy and usability cannot be sufficiently considered by a single type of representation of pseudonyms. Hence, a PIM system should generate digital pseudonyms which are used for communication, while users assign local mnemonics to these pseudonyms in order to simplify their use. We discuss possible solutions for the support of mnemonics and, thereby, propose some improvements to privacy-enhancing identity management tools.

Categories and Subject Descriptors

H.1 [**Models and Principles**]: User/Machine Systems; K.6 [**Management of Computing and Information Systems**]: Security and Protection

General Terms

Human Factors

Keywords

digital pseudonyms, mnemonics, names, privacy, (un)linkability, usability

1. INTRODUCTION

Partitioning person-related data is the only viable approach to maintain privacy despite the fact that information about users necessarily must be acquired in order to provide them a reasonable and useful working environment. Privacy-Enhancing Identity Management (PIM) provides the necessary techniques for this partitioning [3, 4]. Thereby, a data partition represents a partial identity of a user. Such partitioning enables users to be recognized if they act under the same partial identity again. Of course, partial identities must not be linkable except by their owner. Consequently, partial identities of a user can not include a unique attribute such as the user's name.

Therefore, pseudonyms are used as identifiers for partial identities. Generally, a pseudonym can be seen as any identifier (such as a text or bit string) that replaces the real name of a user. There are various kinds of pseudonyms which provide different degrees of anonymity [9]. Since pseudonyms which are used in several transactions enable linkability of all actions performed under this pseudonym, known classifications of pseudonyms mainly consider the frequency of their use: A pseudonym that is used only in a single (trans)action (*transaction pseudonym*) offers the maximum degree of anonymity since it cannot be linked to any other action of its holder. Less anonymity is provided by *role pseudonyms*, *relationship pseudonyms*, and *role-relationship pseudonyms*, which are used for acting in a specific role, for communicating with the same partner, or both. Finally, there are *person pseudonyms* which are a simple substitution of the holder's name and, therefore, offer the least degree of unlinkability.

Additionally to the frequency and scope of use of a pseudonym, we also have to consider by which bit string it is represented, or in other words: the *internal representation* of the pseudonym. Within this paper, we discuss implications of this representation on linkability and, thereby, on anonymity of users. Randomly generated pseudonyms are best suited regarding unlinkability. However, users will only use different pseudonyms and keep them apart if these pseudonyms are presented to them in a way that the pseudonyms are easy to recognize. Users surely do not want to use automatically generated pseudonyms. On the other hand, a user-friendly presentation of pseudonyms surely describes some important aspects of a partial identity.

Hence, users may wish to additionally assign some shorthand description of the semantics — a *mnemonic* — as alias to these randomly generated pseudonyms. A mnemonic is the *presentation* of a pseudonym to a user. It can be of arbitrary type, for example images or sound instead of a text.

As long as mnemonics are solely used locally, i.e., at indi-

DIM'05, November 11, 2005, Fairfax, Virginia, USA.

vidual client sites, their context dependence does not endanger unlinkability. Such limited use seems to be reasonable as long as users interact with their machines only as it would be in typical client-server scenarios. However, in case of interactions between users mediated by their machines, we also have to consider requirements on the presentation of partial identities of others or to others, respectively. Just consider a discussion in a web forum: Even if the same pseudonyms are used again and again, recognizing other users may be quite difficult if the others are identified by their native randomly-generated pseudonyms only. Since usability is essential, the assignment of mnemonics to pseudonyms should be explicitly supported by PIM systems. Within this paper, we want to substantiate this requirement and discuss possible solutions.

In Section 2, we first summarize requirements on pseudonyms and discuss advantages and disadvantages of different (re)presentations of pseudonyms. We compare different approaches to support mnemonics within a system in Section 3. Section 4 discusses challenges and open problems that can not be solved with our approach. Finally, Section 5 summarizes and gives an outlook.

2. DIFFERENT (RE)PRESENTATIONS OF PSEUDONYMS

2.1 Requirements on Pseudonyms

Different requirements on pseudonyms influence their internal and external representation:

(1) A pseudonym by itself must *not leak any information* about the user. Information leakage can threaten privacy since it influences unlinkability. The authors of [9] define unlinkability as follows: "Unlinkability of two or more items of interest (IOIs, e.g., subjects, messages, events, actions, ...) means that within the system (comprising these and possibly other items), from the attacker's perspective, these items of interest are no more and no less related after this observation than they are related concerning his a-priori knowledge." For a more formal definition of unlinkability, we refer to [7].

For our discussion, we have to consider two unlinkability aspects. Firstly, an observer must not gain additional knowledge about the relation of real identities of users and pseudonyms used within the system (*linkability of pseudonym to user*). If pseudonyms by themselves leak information about their holders, this requirement is violated. Secondly, we have to consider *linkability of different pseudonyms*. If users generate different pseudonyms in a similar manner, an observer can link these pseudonyms more easily.

(2) Pseudonyms should reasonably *enable communication* between users. In order to enable reasonable communication between users, users must be recognizable for their communication partners. Some transactions also require *authentication* or *authorization*, respectively.

(3) Finally, we have to consider *usability* of pseudonyms for users. Requirements on usability are, for example, defined by ISO 9241: "The effectiveness, efficiency, and satisfaction with which specified users achieve specified goals in particular environments." [6].

We consider pseudonyms as usable, if they support users

- in managing their partial identities, e.g., in
  - selecting the right partial identity,
  - deciding to generate a new partial identity, or
  - assessing their current linkability, and
- in managing partial identities of other users, e.g., in
  - recognizing former communication partners or
  - managing information they have learned about their communication partners.

Thus, usability supports knowing the details of one's situation regarding privacy and, therefore, enable users to reasonably manage their personal data.

Usually, a pseudonym is considered secure if it fulfills the requirements (1) and (2). Consequently, if requirement (3) is ignored, the implementation will be digital pseudonyms automatically generated by a PIM system and then users will have to use these randomly generated pseudonyms. However, such pseudonyms do not sufficiently support usability which quite probably will undermine security: Users might try to simplify their use, e.g., by assigning alias names they unintentionally could use in communications and thereby divulge additional information. They also could make errors in using such pseudonyms, e.g., choosing wrong pseudonyms. Finally, pseudonyms which are not usable will not establish on a big scale.

2.2 Comparing Different Types of Pseudonyms

The different requirements on pseudonyms imply different types of pseudonyms which can be divided into two groups: *Randomly generated pseudonyms* and *Pseudonyms chosen by users* (Figure 1). Randomly generated pseudonyms, namely *arbitrary random bit strings* and *public keys*, have similar properties. They are generated by a computer randomly and independently within a set and are completely application independent.[1] If we regard randomly generated pseudonyms as abstract data types, the only operations that are possible for this type regarding linkability are $=$ and $\neq$. If two pseudonyms are not equal, they are stochastically independent. An observer cannot draw further conclusions. Particularly, he cannot link different pseudonyms. Even the observation of many of these pseudonyms does not help to identify the user on the long run.

Pseudonyms chosen by users contain some context semantics since users introduce a short-hand description for partial identities in all likelihood. Thus a user could chose pseudonyms which simplify recognizing the context in which he has established the corresponding partial identity or got in touch with it, e.g., the application, the role, the use case, or the communication partner(s).

Considering also this kind of pseudonym as abstract data type, we cannot restrict possible operations regarding linkability on comparison on (in)equality. If two pseudonyms are not equal, an observer might still get information due to similarities between them. In the worst case, the application

[1]Of course, the generation process itself will be deterministic but depend, e.g., on a random seed.

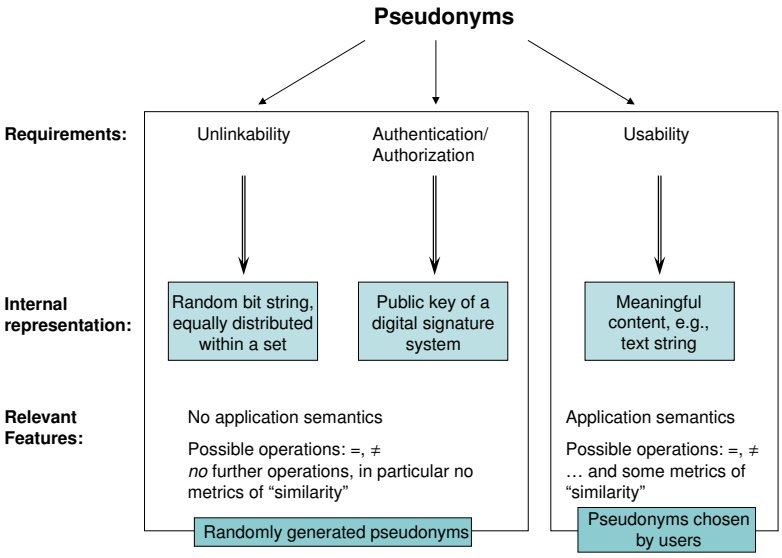

Figure 1: Types of Pseudonyms.

semantics helps him to link or even to partly de-anonymize different pseudonyms. On the long run, he might even be able to identify the holder of the pseudonym.

Even if users generate their pseudonyms independently from the application context, different domains used for choosing the pseudonyms enables linking to a certain degree. At least, we have to consider probabilistic conclusions. For example, one user always assigns forenames to his pseudonyms, another uses names of animals, a third decides to use random numbers of a fixed length. Table 1 compares the different types of pseudonyms regarding linkability and usability.

	Randomly generated pseudonyms	Pseudonyms generated by users
Linkability of different pseudonyms	Unlinkability due to stochastical independence of each other	Features of pseudonym might enable some linkability
Linkability of pseudonym to user	Zero application dependence: observer cannot draw any conclusions about context	Application semantics critical: includes description of context
Usability	Hard to manage partial identities for users	Simplifies managing partial identities for users

Table 1: Comparing different representations of pseudonyms.

Until now, we have discussed the clear distinguished cases "randomly generated pseudonym" and "pseudonyms chosen by users" only. Thereby, we have assumed that the former type does not contain further attributes that might influence privacy. This assumption might not always be true in practice. For example, users can use different security settings in generating their key pairs. Generally, we have to consider linkability of randomly generated pseudonyms if it is not guaranteed that any generated pseudonym could have been produced with the same probability by any of the participating PIM systems. The latter risk can be minimized if all participating PIM systems adhere to the same public rules how to generate these pseudonyms.

We also have to consider that there are possible mixtures between these two types. A user could use his name extended by a random part as public key, e.g., as public exponent within RSA. Obviously, such a key is not context independent. Generally, features of pseudonyms which are visible for others (e.g., the representation, the length, and the domain the pseudonyms are chosen from) must not depend on the user or on properties that should be kept secret. Consequently, these attributes cannot provide usability. Therefore, we have to treat generating the representation and assigning an appropriate presentation separately.

Thereby, the PIM system must generate pseudonyms randomly and stochastically independently from any context information so that they could have been chosen by any of the participating PIM systems with equal probability. Users assign mnemonics to these pseudonyms which support usability. Since we want to prevent possible weaknesses, mnemonics should be explicitly supported by the PIM system. By this, different applications can reuse the functionality for the support of mnemonics. Different possibilities for the management of mnemonics, especially regarding their scope and issuer, are discussed in the following section. For the suggested solution, a PIM-aware platform must be available that provides PIM services to applications. Such a platform is currently developed within the European project PRIME[2] which is the context of our work.

[2]http://www.prime-project.eu.org/

3. ASSIGNMENT OF MNEMONICS

3.1 Global Mnemonics

If the scope of mnemonics is global, they are visible to all users of the corresponding application(s) which simplifies group discussions. As discussed in Section 2.2, we have to assure that attributes of global mnemonics do not depend on user related properties. If we want to ensure that a global mnemonic could be assigned to each user with the same probability, they should be randomly chosen from the same set. Consequently, users cannot select mnemonics on their own. At least, the PIM system would have to check the suitability of a mnemonic chosen by a user. However, as long as IT systems cannot fully understand the meaning of data, this will be simply impossible. Therefore, the selection of mnemonics must adhere to global rules which ensure that the mnemonics are randomly chosen within the defined scope. Particularly, we consider their selection from a *global dictionary*. Several possibilities for this selection are discussed in the following.

Server PIM system assigns global mnemonics (Figure 2). The server PIM system manages the dictionary and assigns mnemonics to pseudonyms of users. First, each PIM client randomly generates a pseudonym. This pseudonym is transmitted during the negotiation phase to the PIM server which assigns a mnemonic to this pseudonym. To simplify matters, we consider the use of a collision-free hash function for selection from the dictionary: This function maps the pseudonym on a mnemonic. At the end of negotiation, the client PIM system gets this mnemonic. If a user wants to communicate with another user, the server mediates this communication using the assigned mnemonic(s) only. Thus, each user knows his own pseudonyms and mnemonics, and mnemonics of other users.

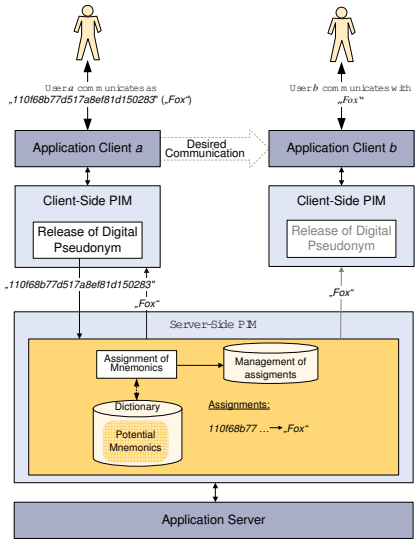

Figure 2: PIM server assigns global mnemonics.

Client PIM system assigns global mnemonics (Figure 3). We assume that the hash function, which is used for mapping pseudonyms on mnemonics, is publicly known. Under this assumption, each PIM client can use this function in order to map randomly generated pseudonyms on mnemonics. For communication, the PIM client uses solely this mnemonic. It learns the mnemonics of the other users as described above during its actions on the server.

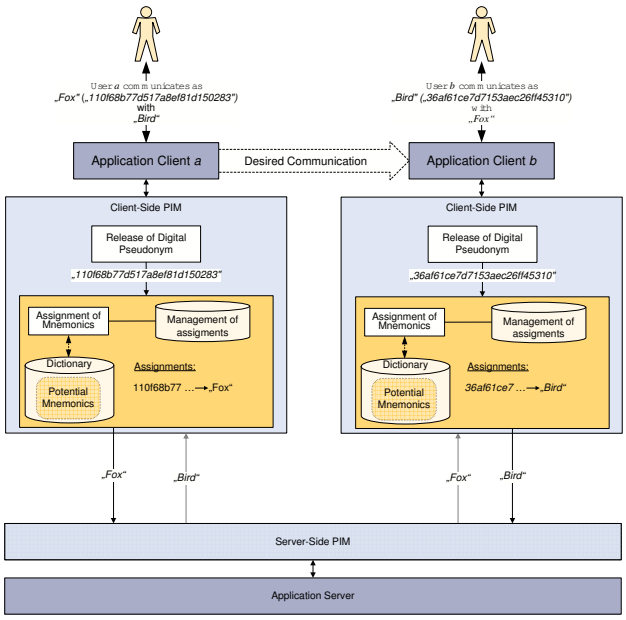

Figure 3: PIM client assigns global mnemonics.

Arbitrary client's PIM system assigns a global mnemonic (Figure 4). Despite these two basic cases, we can also consider other procedures to assign global mnemonics. For example, a PIM client can assign a global mnemonic to a communication partner who does not yet have one (more precisely, to a partial identity that was not assigned a mnemonic before). Consequently, PIM clients can also communicate using their randomly generated pseudonyms. They can receive pseudonyms of others or mnemonics. Additionally, mnemonics could also be assigned by a group of users, e.g., during a discussion: Users may wish to refer to a user who has raised a question or posed a discussion topic.

If we enforce uniform similarity of global mnemonics by means of a global dictionary, mnemonics can be used within communication without the risk to decrease anonymity. However, this global solution has two main drawbacks:

- First, the global dictionary must contain enough mnemonics for all users. If we consider a large application with a huge amount of users, and consider partitioning depending on contexts within this application, we can expect that the necessary size of the dictionary is quite large. Actually, the size may even become unmanageably large because of the birthday problem [5].[3] In the worst case, it will not be possible to find enough distinct — and user-friendly — mnemonics.

[3]For a relatively small set of pupils the probability that two of them have the same day of the year as birthday is surprisingly great. Even if relatively many mnemonics are possible, the probability that two users will be assigned the same one is not negligible.

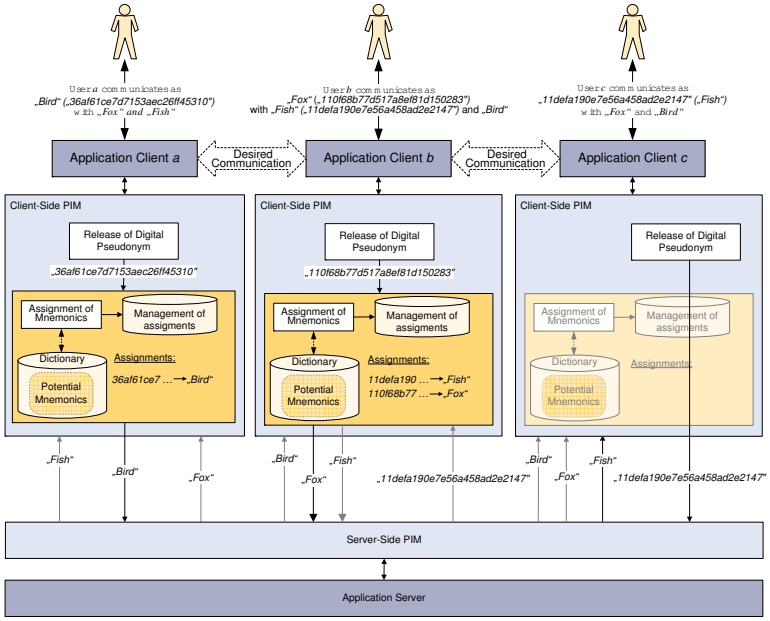

Figure 4: Arbitrary PIM client assigns global mnemonics.

- Second, global mnemonics might not actually meet the requirement on usability. Mnemonics are only useful if users are familiar with them. For example, cultural as well as linguistic differences influence their usability. Therefore, users may wish to assign own mnemonics.

The next section investigates local assignment of mnemonics which is a promising way to overcome the problems of global assignment.

3.2 Local Mnemonics

In this alternative, PIM clients assign local mnemonics to their own pseudonyms as well as to pseudonyms of other users (Figure 5). Within communication over the network, only the randomly generated pseudonyms are used. Local assignment of mnemonics does not bear the risk that there are not enough meaningful mnemonics. Furthermore, each user can adapt the mnemonic to his own preferences. Obviously, this solution provides the best usability.

Each particular mnemonic shall only have meaning within the domain of a single client side. This means that the same mnemonic may be a mnemonic to different pseudonyms at different client sides as well as that a local mnemonic should not be communicated to others: Firstly, because they have no well defined meaning to them anyway. Secondly, because the way mnemonics are chosen may give person-related information to others which could be used to link pseudonyms.

Table 2 compares some features of global and local assignment of mnemonics. Global assignment provides advantages regarding their usage within communications. However, local assignment provides advantages regarding scope and usability. Therefore, we prefer local assignment.

These arguments suggest that the following conclusions should be considered in all implementations of pseudonyms:

> Instead of letting users choose pseudonyms that are both secure and usable, the PIM system should generate secure internal representations of pseudonyms which are used for communications between machines. To make the pseudonyms usable for each individual user, each one assigns mnemonics of his choosing to achieve also usability.

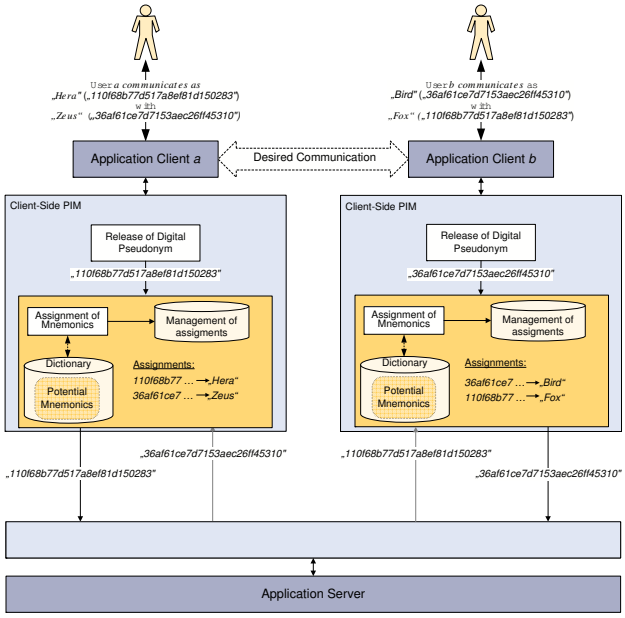

Figure 5: PIM client assigns local mnemonics.

	Global Assignment	Local Assignment
Scope	Whole system, which requires a global dictionary that must include enough entries.	Domain of client only — easier to find enough meaningful mnemonics.
Use within communication	Can be used without any restrictions.	May not be used since this would disclose person-related information.
Usability	It is questionable whether global mnemonics are really helpful for all users of a large system.	Users can define mnemonics according to their preferences which ensures the best usability.

Table 2: Comparing global and local assignment of mnemonics.

4. CHALLENGES AND LIMITATIONS

According to the discussion in Section 3, we propose to use locally assigned mnemonics. Currently, we are working on an implementation for an example application within the project PRIME [2]. The application uses services provided by a PIM system developed within this project in order to support users in partitioning their personal data [1]. Users should be enabled to assign suitable mnemonics and to decide by means of information about their current privacy state as well as about former actions whether a switch to another partial identity or the generation of a new partial identity is necessary.

If users want to address other users, either directly or indirectly in a discussion with other members of the communication channel, it is tempting to use the mnemonic as pseudonym. Since locally assigned mnemonics should not be mentioned in communications, the PIM system should:

1. replace them by the pseudonym before storing a document or transmitting a message, and
2. present all pseudonyms by the mnemonics defined in the domain of their current presentation.

Hence, mnemonics should be entities administered by the local PIM component and known to, e.g., the text-processing software used. Since typos could cause problems for this replacement, users need reasonable support in addressing other users.

A privacy-aware interface is required for this task. The application user interface needs to be enriched by information about the user's current privacy state. So it should inform a user about the pseudonym currently used and about information that is linked to this partial identity. Furthermore, the interface should also inform the user about the partial identities of other users. This presentation should also support users in contacting other users as described above. All these extensions improve usability for users.

Generally, users might unintentionally disclose information about them due to the content of transmitted data. Evaluating the content of messages and providing reasonable assistance is extremely difficult as long as the semantics of data cannot be automatically understood. It is quite difficult to assist users in not providing information about them within usual discussions: Any topic can be discussed, users can make any remarks. They can either directly disclose information, or they can make remarks which allow other humans to draw conclusions about them. The latter depends on the a priori knowledge of the other users. To conclude, the problem is very difficult and it cannot be expected to have a general solution.

Additionally to the issues discussed so far, there are further threats that we have to consider. Even if we choose and apply pseudonyms as well as mnemonics very carefully, we cannot prevent that information leaks via the actions themselves. Particularly, an observer could analyze the content of messages in order to assign them to their sender. Typical typos as well as preferred wording could help to identify the issuer of messages or at least to link different messages in a first step. The authors of [10] perform syntactic and semantic analysis considering classical stylometry in order to assign messages to their authors. In empirical tests, they have found out that reasonable analysis require about 6500 words posted by users. Improved results yield analysis described in [8].

From these analysis, we conclude that a fine-grained partitioning of personal data becomes even more important: The less data delivered by a user an observer can analyze, the worse are his chances to identify the user. Consequently, the PIM system has also to consider the amount of data delivered by a user and inform him if he should switch to another partial identity.

5. SUMMARY AND OUTLOOK

This paper has discussed possible impacts of the (re)presentation of pseudonyms on the achievable privacy. Usability is an important aspect of pseudonyms. Users will only make use of possibilities to partition personal information if the use of this technique is comfortable for them. Particularly, they will surely not introduce many different partial identities if recognizing these identities is difficult. Therefore, the use of mnemonics as usable presentation of secure pseudonyms should be explicitly supported.

We conclude from our discussions, that a PIM system should generate digital pseudonyms and enable users to assign mnemonics to pseudonyms. Mnemonics should be used at individual client sides only. This local scope provides the best usability while preventing possible privacy problems which might result from the application dependence of mnemonics chosen by users. In order to prevent that users unintentionally disclose information about their pseudonyms, their communication tools must be adapted. For example, messages must be parsed in order to replace mnemonics by pseudonyms.

Further research has to be done in order to develop suitable support of users in assigning mnemonics. Furthermore, it is planned to use the developed tools in different trials in order to assess the achieved enhancements of PIM systems. Of course, usability is an important aspect of these trials.

Another topic of future research is to investigate alternative mnemonics for pseudonyms: Firstly, multimedia based representation further increases usability of pseudonyms. Secondly, representation of information can be adapted in order to support disabled users.

Another reason is the necessity to adapt the mnemonic to different end devices, especially if we consider mobility of users. Thereby, we also have to consider impacts of adapt-

ing the presentation on fairness as well as on privacy. Particularly, these impacts become important in case of synchronous group communication. Depending on the chosen presentation and on the computing power of the end device, adapting can require some effort and cause delays for the user which might make it difficult for him to participate on a communication.

If he contributes anyway, other users can recognize that he always answers quite late, possibly if they have already started to discuss a new topic. Finally, they can draw conclusions about this user: Either this user necessarily needs an adaptation on his end device, which might give hints about a disability, or he uses a quite slow end device which might give hints about his current working environment or budget. Thereby, we assume that users would deactivate adaptations which cause delays if possible.

Investigations to utilize multimedia based presentation of mnemonics have to consider these problems in order to prevent impacts on privacy. The first simple solution would be to consider a very efficient presentation for each end device. In case of synchronous group communication, the users can negotiate minimum and maximum delay times and use the efficient presentation if necessary.

6. ACKNOWLEDGEMENT

We like to thank Mike Bergmann, Sebastian Clauß, Thomas Kriegelstein, Katja Liesebach, and Hagen Wahrig for fruitful discussions. 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 his/her sole risk and liability. The work reported in this paper was supported by the IST PRIME project; however, it represents not necessarily the view of the project. The PRIME project receives research funding from the European Community's Sixth Framework Programme and the Swiss Federal Office for Education and Science.

7. REFERENCES

[1] K. Borcea, H. Donker, E. Franz, K. Liesebach, A. Pfitzmann, and H. Wahrig. Intra-application partitioning of personal data. 2005. Workshop on Privacy-Enhanced Personalization (PEP 2005).

[2] K. Borcea, H. Donker, E. Franz, A. Pfitzmann, and H. Wahrig. Towards Privacy-Aware eLearning. 2005. Proceedings of PET 2005.

[3] S. Clauß and M. Köhntopp. Identity management and its support of multilateral security. *Computer Networks*, (37):205–219, 2001.

[4] S. Clauß, A. Pfitzmann, M. Hansen, and E. V. Herreweghen. Privacy-enhancing identity management. *The IPTS Report 67*, pages 8–16, September 2002. http://www.jrc.es/pages/iptsreport/vol67/english/IPT2E676.html.

[5] D. W. Davies and W. L. Price. *Security for Computer Networks*. John Wiley & Sons, 2nd edition, 1989.

[6] A. Dix, J. Finlay, G. Abowd, and R. Beale. *Human Computer Interaction*. Prentice Hall Europe, 1998.

[7] S. Köpsell and S. Steinbrecher. Modelling unlinkability. In *Workshop on Privacy-Enhancing Technologies (PET) 2003, LNCS 2760*, pages 32–47. Springer-Verlag Berlin, 2003.

[8] J. Novak, P. Raghavan, and A. Tomkins. Anti-aliasing on the web. In *Proceedings of the 13th Int. Conference on the World Wide Web*, pages 30–39, 2004.

[9] A. Pfitzmann and M. Hansen. Anonymity, unlinkability, unobservability, pseudonymity, and identity management — a consolidated proposal for terminology. http://dud.inf.tu-dresden.de/Anon-Terminology, August 2005.

[10] J. R. Rao and P. Rohatgi. Can pseudonymity really guarantee privacy? In *Proceedings of the Ninth USENIX Security Symposium*, pages 85–96. USENIX, Aug. 2000.