Privacy and Identity Management in Europe for Life

Final Report on Analysis, Scenarios, Requirements and Concepts, Evaluation Results of "Privacy in Life"

Editors:	Katrin Borcea-Pfitzmann (TUD)
	Jan Camenish (IBM)
	Bibi van den Berg (TILT)
Reviewers:	Laurent Bussard (EMIC)
	Erik Wästlund (KAU)
Identifier:	D1.3.3
Type:	Deliverable
Version:	1.1
Class:	Public
Date:	June 14, 2011

Abstract

Activity 1 of project PrimeLife deals with the development of prototypes and demonstrators dedicated to selected important research fields of "Privacy in Life". This deliverable provides a comprehensive summary of the results elaborated within the three work packages of Activity 1.

This includes, firstly, aspects of privacy in social software comprising audience segregation in social network sites, securing message exchange in social settings as well as privacy-preserving access control in collaborative workspaces. Secondly, concepts for privacy-preserving user reputation and trustworthiness of user-generated content are discussed and developed prototypes are described. And thirdly, a comprehensive analysis of peculiarities and requirements related to lifelong privacy management is given. Further, an overview of the developed prototype addressing lifelong aspects as well as the main results of the evaluation of the prototype are provided.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 216483 for the project PrimeLife.

Members of the PrimeLife Consortium

1.	IBM Research GmbH	IBM	Switzerland
2.	Unabhängiges Landeszentrum für Datenschutz	ULD	Germany
3.	Technische Universität Dresden	TUD	Germany
4.	Karlstads Universitet	KAU	Sweden
5.	Università degli Studi di Milano	UNIMI	Italy
6.	Johann Wolfgang Goethe - Universität Frankfurt am Main	GUF	Germany
7.	Stichting Katholieke Universiteit Brabant	TILT	Netherlands
8.	GEIE ERCIM	W3C	France
9.	Katholieke Universiteit Leuven	K.U.Leuven	Belgium
10.	Università degli Studi di Bergamo	UNIBG	Italy
11.	Giesecke & Devrient GmbH	GD	Germany
12.	Center for Usability Research & Engineering	CURE	Austria
13.	Europäisches Microsoft Innovations Center GmbH	EMIC	Germany
14.	SAP AG	SAP	Germany
15.	Brown University	UBR	USA

***Disclaimer**: 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 below referenced consortium members shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials subject to any liability which is mandatory due to applicable law. Copyright 2009, 2010, 2011 by IBM Research GmbH, Unabhängiges Landeszentrum für Datenschutz, Technische Universität Dresden, Stichting Katholieke Universiteit Brabant, Katholieke Universiteit Leuven, Università degli Studi di Milano.*

List of Contributors

Contributions from several PrimeLife partners are contained in this document. The following list presents the contributors for the chapters of this deliverable.

Chapter	**Author(s)**
Chapter 1: Introduction	Stefanie Pötzsch (TUD), Sandra Steinbrecher (TUD), and Katrin Borcea-Pfitzmann (TUD)
Chapter 2: Privacy in Social Software	Bibi van den Berg (TILT), Stefanie Pötzsch (TUD), Filipe Beato (K.U.Leuven), Eros Pedrini (UNIMI), Tim Holweg (TILT), Coen van den Munckhof (TILT), Yuanhao Sun (TILT), and Joeri de Ruiter (TILT)
Chapter 3: Trustworthiness of Online Content	Jan Camenisch (IBM), Sandra Steinbrecher (TUD), Ronald Leenes (TILT), Stefanie Pötzsch (TUD), Benjamin Kellermann (TUD), and Laura Klaming (TILT)
Chapter 4: Identity and Privacy Issues Throughout Life	Stefan Köpsell (TUD), Jaromir Dobiáš (TUD), Arnold Roosendaal (TILT), Marit Hansen (ULD), Maren Raguse (ULD), Leif-Erik Holtz (ULD), Ulrich König (ULD), Katalin Storf (ULD), Harald Zwingelberg (ULD), Andreas Pfitzmann (TUD), Sandra Steinbrecher (TUD), and Katrin Borcea-Pfitzmann (TUD)
Chapter 5: Conclusion and Outlook	Katrin Borcea-Pfitzmann (TUD), Jaromir Dobiáš (TUD)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the course of information technologies changing from being “just” a means to store and handle data to a medium enabling social interaction, those technologies become very popular and successful: Individuals exchange and share information of personal and non-personal nature. Those interactions take place whenever they desire and wherever they are, thus, allowing to enormously reduce response times and distances.

However, recent events have shown that such kind of interaction may offend the individuals’ autonomy. Thus, users need support to protect their privacy and to retain control over their personal information, irrespective of their activities. This means that users of information technologies always have to be aware of which data they are producing explicitly and implicitly. They need to know who their audience is and they have to decide whether to trust the members of their audience or not. Information technologies need to adopt the measures for establishing trust we have in the offline world (e. g., usage of ID cards, certifications of rights, word-of-mouth, ...). This raises additional privacy risks for individuals as the information about them for establishing trust needs to be strongly associated with them. Here, a paradox related to online interaction becomes apparent. Online interaction means that more traces are left and that individuals can be recognised easier, while, on the other hand, the physical distance between interacting parties makes it more difficult for them to determine exactly with whom they are interacting.

Activity 1 of the project PrimeLife especially aimes at providing scalable and configurable privacy and identity management in new and emerging Internet services and applications such as virtual communities and collaborative applications. Accordingly, demonstrators have been set up, firstly, showing how audience segregation for data can be realised as documented in Chapter 2. Secondly, in Chapter 3, we describe trust mechanisms to help users decide on trustworthiness in data delivered by others, with all users concurrently having privacy requirements on the data they request and deliver. The scenarios we chose for audience segregation and trust management cover a large part of Web 2.0 applications (blogs, wikis, social networks, forums). The prototypes we built for these scenarios served as a basis for experiments for finding out which indicators raise users’ awareness with respect to data’s privacy and trustworthiness. Most of these tools are also available as open-source for further development and usage.

Another longer-term goal of Activity 1 is to provide individuals with means to protect their privacy over their whole lifespan (cf. Chapter 4). Each individual leaves a multitude of traces during a lifetime of digital interactions. The total of these traces forms a digital footprint. While parts of these traces are unconsciously left behind and not meant as important information, a lot of information is very relevant and important for specific domains. This data contains information about an individual, but also all sorts of documents have to be accessible by someone at any time. This implies that, in case an individual is not able to manage this information, others need to obtain access. The inability to manage information can be either temporary, such as during the case of illness, or permanent, when the individual deceases. As an exemplary demonstrator, we built a privacy-preserving backup tool that takes into account new ways of interacting. The tool provides for backup and synchronisation of data. With a view on the specific requirements concerning privacy and identity management, mechanisms are built in to safeguard individuals, to improve control over the data and to allow for delegation.

Since this deliverable reports on the main results of Activity 1 of the PrimeLife project, a large part of this deliverable's content has been published in form of the PrimeLife book "Privacy and Identity Management for Life" [CFHR11]. However, this deliverable also reports on new findings and developments achieved after the finalisation of the PrimeLife book.

Chapter 2

Privacy in Social Software

2.1 Introduction: Privacy Issues in Social Software

In recent years ‘web 2.0’ has become a commonly used term to describe the latest generation of the Internet. The term refers to four general characteristics of the new Internet, which set it apart from the earliest days of the web – now loosely called ‘web 1.0’. These four characteristics are:

1. Internet users are no longer merely consumers of information on the Internet, but have an increasingly important role as *creators* and *distributors* of content [Tap09, How08, Lea08];

2. Web 2.0 offers a wealth of new channels for communication and interaction. Social interaction, therefore, is one of the key parameters of this new generation of the Internet, which is why web 2.0 is often called the ‘*social web*’;

3. In a variety of web 2.0 domains users cooperate to create or distribute content, which is why *participation* and *co-creation* are also key terms for the new Internet;

4. Technically, web 2.0 differs from the first generation of the Internet in the sense that much of the software that users access to share, distribute and download content, or engage in interaction, is *placed on the Internet*, and is accessible through any browser. The use of such software is provided as a service, and the Internet has become the central platform for users to access different types of software [O’R07].

Mergel *et al.* summarise these four characteristics in an oft-quoted definition of web 2.0. According to them, web 2.0 is a

> “...set of economic, social, and technological trends, that collectively form the basis of the next generation of the Internet – a more mature, distinct medium characterised by user participation, openness, and network effects” [MSF09].

As said, web 2.0 is often called the ‘social web’. This refers, on the one hand, to the fact that social interaction has become a central element of users’ behaviour on the Internet. Users connect with others, both known and unknown to them, via forums, social network sites, on twitter, in wikis, via blogs, and so on and so forth. On the other hand, the term ‘social web’ also refers to the *software* used in web 2.0, which aims at social ends. This software is collectively called ‘social software’. Richter and Koch [Ric07] have shown that social software has roughly three functions:

1. *Information management*: individuals can use social software to find, evaluate, disseminate, and administer information. Examples of this functionality include the video-sharing domain YouTube, the Internet encyclopaedia Wikipedia, and forum software;

2. *Management of self*: individuals can use social software to present (aspects of) themselves on the Internet, i.e. to create, express and manage their online identities. Examples of this functionality include social network sites, blogs and twitter;

3. *Relationship management*: individuals can use social software to engage in social interaction with others, both known and unknown in the ‘real’ world, via the Internet. This type of functionality is most explicitly present in social network sites, twitter and blogs, but forums and other social domains on the Internet may also be used for this purpose.

The success of web 2.0 builds on the active participation of users, and on their willingness to contribute to the creation, dissemination and evaluation of content [SGL06, O’R07]. However, when individuals use social software, an extensive amount of personal data may be disclosed – either deliberately or accidentally. Moreover, such data can be disclosed directly, for instance, when individuals share their real name and birth date via a social network site, or indirectly, as is the case, for instance, when users edit a topic in a wiki, comment on a blog entry or post a statement in a forum [GA05, EGH08].

In one of the workpackages of the PrimeLife project we set out to create an inventory of the possible threats to privacy that are generated by the use of social software. We began by conducting an extensive literature review of articles and books that discussed both theoretical work on, and empirical studies of, privacy issues in web 2.0. Using these materials, we conducted a broad analysis of the various privacy threats to individual users in relation to social software. The results of this analysis were discussed in great detail in earlier heartbeats and deliverables[1], and were also published in various shorter articles and book chapters[2]. Therefore, we will only summarise them briefly here.

We started out by making a distinction between two general categories of social software in which privacy issues emerge: (1) *social network sites*, e.g. Facebook, LinkedIn, Friendster, and so on and so forth, and (2) a category that we’ve collectively called ‘*collaborative workspaces*’, which includes all web 2.0 domains in which the primary aim is to create or share content, e.g. forums, wikis, the YouTube video sharing site and so on and so forth. While privacy issues have been investigated quite thoroughly and extensively in relation to social network sites [GA05, db08a, db08b, Gri08, BK09], they

[1] See for example: Deliverable 1.2.1 [WP110a]

[2] See for example [VdBL10, VdBL11, VdBPL$^+$11]

have received far less attention in relation to collaborative workspaces. In our analysis of the landscape of privacy issues in social software, therefore, we have explicitly attempted to widen our discussion to also include the latter area.

Broadly speaking, there are five fundamental issues with respect to privacy risks and the use of social software[3]:

1. When users go online to share content through social software, they lack an overview of who has access to this content. Users cannot adequately judge the size and makeup of their *audience* [PD03, Tuf08]. This is problematic, especially in web 2.0 domains in which users share detailed personal information, for example in social network sites. The key goals of social network sites are to express one's online identity, and to (re)connect with friends and acquaintances or engage in new social relations. Because of these goals users share a wealth of personal information on them, and social network sites actively promote this sharing [Gri08]. However, it is often unclear to users who has access to all this information, and hence privacy issues are a real risk. In forums and other collaborative workspaces, in which users share information and co-create content, privacy issues may also arise, for the same reason. While users may not reveal as much detailed personal information in such domains, by participating in them, they may nevertheless leave traces that enable others to find out information about them, or even trace back information to their natural person. The storage of IP addresses with posts in wikis and forums is an example of this.

2. Information and communication technologies enable anyone to *collect, copy, link and distribute the (personal) data of others*, thus allowing for the creation of extensive profiles of individuals. Information may also easily be copied and stored outside the original domain, thus making it even harder for users to know who has access to their information [Hou09]. This, too, is an issue both in social network sites and in collaborative workspaces. When users post content in collaborative workspaces, such content may be copied to different contexts, thus finding its way to unintended audiences (in one of the scenarios we present below exactly this problem is described). Also, the meaning of the content may change considerably in the process of copying it from one context to another. Similarly, when users post personal information on a profile page in a social network site, this information can easily be copied and stored elsewhere – including outside the network – where it may be accessible to a much wider audience than the user had originally intended, and possibly also altering or obliterating the original meaning of the content.

3. Information and communication technologies facilitate *data storage for a nearly indefinite time period*, thus making it impossible to erase this information [MS09]. Combined with the previous point – that information can easily be copied, linked and stored – this causes significant privacy risks indeed in all types of social software. It means that, even if a user removes an item of information from her profile

[3] Privacy issues caused by third party businesses and by the providers of the social network sites themselves are a serious threat. These types of privacy issues were discussed in Heartbeat 1.2.5 of the PrimeLife project. In this text we have chosen to focus on privacy issues amongst users only, since those are the issues that we primarily attempted to solve in building our three demonstrators, which we will discuss below.

page on a social network site, or content from a collaborative workspace, it may still be found elsewhere on the Internet, and for an indefinite time period. Users thus have even less of an overview over the 'audience' to which they are disclosing information, since the content may be visible to audiences *in the future* as well [PD03].

4. Using social software, anyone can publish personal information about another individual. While active user participation and content creation are some of the main drivers of the new generation of the Internet, at the same time they are also its biggest pitfalls. The fact that any user can publish any kind of information about other persons (either personal or not, and either true or false), may have serious consequences for these persons' reputations [Sol07]. The breadth and seriousness of this problem has been revealed through numerous incidents in both collaborative workspaces and social network sites already, some of which we've discussed in Deliverable 1.2.1 [WP110a].

5. Individuals' lack of *privacy-awareness* when using social software may lead to information leaks and leaving unintended and/or unobserved virtual traces. The previous points have revealed that users have limited *control* over the (personal) information they post in web 2.0 environments. On top of that, however, users are unaccustomed to thinking about the (unintended) audiences that may see the content they post, and the (long-term) effects this may have on their 'real' lives.

Using this set of five central threats to privacy in social software, the PrimeLife project aimed to develop a number of *requirements* that ought to be met in social network sites and collaborative workspaces in order to improve users' privacy protection in using these web 2.0 domains. In the next section we will stipulate what these requirements consist of. But before turning to that discussion, we will first describe two scenarios – one from a social network site and one from a collaborative workspace – to illustrate the discussion of privacy issues in this section.

2.2 Scenarios and Requirements

In the previous years a number of privacy incidents have been reported in relation to the use of social software, both in scientific research and in the popular press. In almost all cases, one or more of the five issues discussed in the previous section were at the heart of the problem. In this section we begin by presenting two such cases, exemplifying the issues that may arise in social network sites and collaborative workspaces – in this case a forum – respectively.

2.2.1 Scenario 1: A Social Network Site

Natalie Blanchard was a member of the biggest social network site in the world: Facebook. She had a profile page, detailing information about her person, her hobbies and preferences. Through Facebook she stayed in touch with a number of contacts, both close friends and acquaintances. Natalie knew that users must be careful about sharing their personal information in social network sites because of privacy issues, so she had changed the privacy settings of her profile to 'visible to friends only'. This means that only the members of her contact list could see the information she posted there. Natalie regularly posted pictures on her profile page, for instance of a trip she took to the beach, or of parties that she attended with friends.

One day in 2009 Natalie received a message from her insurance company, telling her that they were going to terminate the monthly payments she had received for the last year and a half because she was on sick leave – she had been diagnosed with depression in 2007. Inquiries revealed that the insurance company had used Natalie's Facebook page to investigate the validity of her ongoing claim for monthly payments, and had used the pictures of her, happy at the beach or laughing with friends, to conclude that Natalie was unjustly receiving these payments. It remains unclear how the insurance company got access to the profile page, if Natalie had indeed shielded it from everyone who was not on her contact list, as she claims to have done.

This scenario shows that *unintended audiences* may sometimes gain access to personal information in social network sites, and thus receive information that was not posted with them in mind. Based on what they see there, these unintended audiences may draw conclusions, and even undertake actions, that may harm the individual involved in her 'offline' life[4]. Sharing information without having a clear grasp of the makeup and the extent of the audience, this scenario reveals, can have serious repercussions for users.

2.2.2 Scenario 2: A Forum

Hannes Obermaier works as a salesman in a big electronic company and his favourite hobbies are his family and gambling. He plays poker well and has even won a small amount of money in an online poker room. Unfortunately, Hannes forgot to indicate this earning in his tax declaration and therefore he has a problem with his tax office. Seeking advice in this situation, Hannes finds a forum on the Internet where all kinds of questions related to online gambling are discussed. Hannes hopes to find help and creates a forum post in which he describes his problem. After a few minutes, another forum user has written the first reply to Hannes post saying that he has experienced similar problems and asking about some more details of Hannes' case. During the next few days, Hannes spends a lot of time in the forum. He has gotten to know a lot of the other users and with three of them he really feels like they have been friends for ages. Of course, Hannes has told them not only about his problem with the tax office, but he also shared some funny stories from his everyday life and posted a link to a cute picture of his son from his personal homepage.

One day, a new user who calls herself WendyXY appears in the forum and starts to post insults and allegation about Hannes, not just once but repeatedly. The scary thing is that she seems to know Hannes' name, where he lives and for which company he works. Later, Hannes realises that WendyXY may have found his personal homepage since he had posted the link to the picture of his son. His personal homepage contained Hannes' real name and his residence. Knowing this information, it must have been easy

[4] For the purpose of readability, this document refrains from using gender-neutral pronouns such as "he/she". Accordingly, gendered pronouns are used in a non-discriminatory sense and are meant to represent both genders.

for WendyXY to infer where he works since Hannes has briefly mentioned his job in earlier posts and there is only one big electronics company in his local area.

The story becomes even worse when one of Hannes' major clients finds all the allegations about him on the Internet and cancels an important contract for fear of a negative image.

This scenario may seem artificial, yet a similar case was reported by the German newspaper *Zeit* in 2009 [Bur09]. It illustrates that sharing of personal data with possibly millions of unknown people on the Internet is a critical point from a privacy perspective and may result in negative consequences, such as bullying, cyberstalking or harassment. However, note that the sharing of personal data with an *intended* audience – in this scenario for example talking about the tax office problem with other online gamblers – is the main reason to use forums or similar collaborative workspaces.

2.2.3 General Requirements

When studying the five privacy risks in using social software that we've discussed above, and combining these with the illustration of such risks in real-life cases, as described in the scenarios, we draw the following general conclusion: privacy problems in using social software often revolve around '*context collision*' [RG10], because information is disclosed to *unintended audiences*.

The term 'context collision' refers to the fact that in these web 2.0 domains the various separate social spheres that individuals have (and appreciate!) in their 'real' lives are flattened, merged, and conflated . For example, in many social network sites *all* the information that users post on their profile page is accessible to their contacts (and sometimes even to all members of the entire network); users cannot choose to hide certain items of content from specific persons or groups, for example keeping family pictures for other family members only. As Raynes-Goldie remarks quite rightly:

> "As a result a user's teetotaller boss sees the same things as their best friend, the party animal. This can cause problems when trying to decide what to share about yourself, or trying to manage how people from different life contexts might perceive [information]. What is appropriate for a user's friends to see may not be appropriate for their employer" [RG10].

Context collision means that individuals cannot meet the various behavioural requirements of the different social settings in which they normally operate, since one and the same audience sees all of their behaviours. The same applies to collaborative workspaces, as we have seen in the scenario of Hannes Obermaier, in which information from various contexts collided and led to serious problems in both his online and his offline life.

Another way of framing the issue of context collision is to say that when using social software, users lack means for *audience segregation* [Gof59]. The twentieth-century Canadian sociologist Erving Goffman, who coined this term, pointed out that human beings need such a segregation of audiences, since they perform different, possibly conflicting, *roles* throughout their everyday lives, and the impressions they aim to foster in each of these roles must not be contaminated by information from other performances. With segregated audiences for the presentation of specific roles, performers can 'maintain face' before each of these audiences. Due to the five fundamental issues stipulated in the previous section, maintaining separate audiences is very complicated in web 2.0 environments, and this, as we have seen, entails significant privacy risks.

We conclude that (a) audience segregation is a viable and an important means of protecting individuals' privacy in the offline world, and that (b) this mechanism is sorely lacking in current social software environments – both in social network sites and in collaborative workspaces. Based on this conclusion, in PrimeLife we set out to create *virtual means of segregating audiences* in three different demonstrators. When translating the notion of audience segregation to technical terms, we soon realised that the core mechanism to strive for was the creation of tools for *selective access control* in both social network sites and collaborative workspaces. The top-line requirement we set out to realise for the various types of social software under review was to create different ways of activating, managing, and improving selective access control.

However, when looking at the different categories of social software, we soon realised that each has different, more specific requirements, due to the differences in their makeup and the key goals for which that have been created. Social network sites require different mechanisms for audience segregation than collaborative workspaces.

In social network sites users are directly connected to other members, whom they (mostly) know (at least to some degree). This means that generating selective access control in social network sites refers mainly to the fact that users must be able to make information visible to specific *contacts*, yet not others. This can be realised by enabling users to create multiple profile pages in a social network site, and to cluster contacts in subgroups so that information disclosure becomes more targeted and precise. We will outline these principles in more detail in the next section when we discuss our social network site demonstrator called Clique.

In collaborative workspaces, such as forums, users are generally not connected to others, and they may not know any of the other members of the workspace. In these collaborative workspaces privacy-enhancing selective access control can be realised based with respect to the *general properties of the intended audience*, without having to 'know' each other user in particular. This principle will be discussed in more detail in section 2.6, where we discuss the forum demonstrator that we've developed.

2.3 Three Prototypes to Improve Privacy in Social Software

To investigate and exemplify the way in which audience segregation (i.e. selective access control) could be implemented in the types of social software we've discussed, we built three demonstrators in total. The first demonstrator is an instance of a privacy-enhanced social network site, in which a number of mechanisms for audience segregation are implemented. The second demonstrator is an encryption tool called Scramble! that can be used in social network sites, but also in other web 2.0 domains. It is a Firefox plugin that can encrypt any content that users type into fields on the Internet. The final demonstrator is a mechanism for selective access control in a collaborative workspace, more specifically in a forum. We've discussed these demonstrators, their technical details and their practical workings in several previous heartbeats and deliverables[5], and in

[5] Most importantly, see: Deliverable 1.2.1 [WP110a] and Deliverable 1.2.2 [WP110b]

multiple publications[6]. We've also illustrated them with screenshots in these documents. Therefore, in this deliverable we've chosen to only briefly discuss an overview of each, and to focus instead on the work we've done in the year 2010-2011 to improve the earliest versions of each of them, based on user studies, on feedback from peers after presenting them at conferences, and on progressive insight in the respective research communities they've originated from.

2.4 Clique: Contributing to Solving Privacy Issues in Social Network Sites

To tackle some of the privacy issues we've discussed above in relation to the use of social network sites we developed our own 'alternative' social network site called 'Clique'. Clique was built using Elgg open source software. It was developed in the winter of 2009 and early spring of 2010 and went online in March of that year[7]. We had originally intended this demonstrator as precisely that: a proof of principle only. However, soon after its launch Clique received considerable media attention, and this has led to a far larger group of (active) users than we had originally foreseen. Since Clique was intended as a demonstrator only, we aimed at a group of approximately 40-60 users – mostly consisting of PrimeLife researchers and some privacy-interested colleagues outside that group. However, since its launch Clique has gathered approximately 1.800 users[8] from a number of different countries, both within Europe and outside, and from a variety of backgrounds and ages.

2.4.1 Audience Segregation in Clique: Three Mechanisms

To realise audience segregation in Clique, we created three different mechanisms, which we will discuss in turn in this section:

1. Contact management through the use of '*collections*',
2. Setting visibility rights and
3. Managing multiple '*faces*' to show different sides of oneself to different audiences.

Collections. The first mechanism we implemented was that of *enabling contact management through the use of 'collections'*. As has become clear from our discussion of audience segregation above, this social mechanism is based on nuances in connections [Db04]. This means that users are able to create their own social clusters (which we have called 'collections'), in which they group one or more of their contacts, and that they can assign labels to these clusters. This departs from most current-day social network sites, in which all contacts in a user's network are lumped together in one cluster of 'friends'. By allowing users to create collections within their list of contacts, they can cluster social relations according to their own preferences, thereby mimicking the

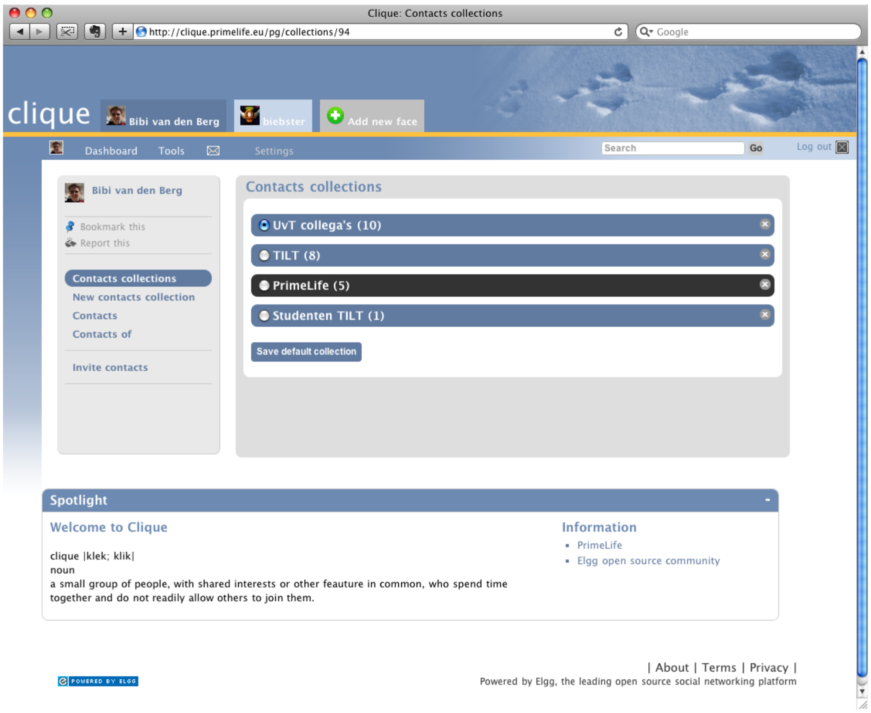
[6] See for example [VdBL10, VdBL11, VdBPL$^+$11]

[7] http://clique.primelife.eu/

[8] At the date of writing this deliverable, i.e. in April 2011

actual practice of building and maintaining separate social spheres in real life. Users must be free to define (and label) their own collections, since that is the only way in which these collections will correspond to the fabric of their social life. Grimmelmann [Gri08] has argued that if the provider of the social network site offers the possibility to place contacts in clusters (such as 'family' or 'friends'), these clusters could never be an adequate representation of the complexity of social relationships in real life. He is correct in this observation. Platform providers could never capture the complexity of individuals' many social spheres and connections. However, in Clique we aimed to show that the individuals *themselves* are fully capable of doing so. We all know which individuals make up our social circle and what the different degrees of intimacy in that social circle consist of. In Clique, therefore, we have built a tool for contact management that allows users to replicate their social sphere in any level of granularity that works for them. This solves the problem signalled by Grimmelmann.

In Clique users can cluster contacts into self-assigned and self-labeled sets. After inviting contacts, they are asked to assign them to one or more 'collections', which can be changed at any time. Figure 1 shows the management of collections in Clique.



clique |klek; klik|
noun
a small group of people, with shared interests or other feauture in common, who spend time together and do not readily allow others to join them.

Powered by Elgg, the leading open source social networking platform

Figure 1: Collections in Clique.

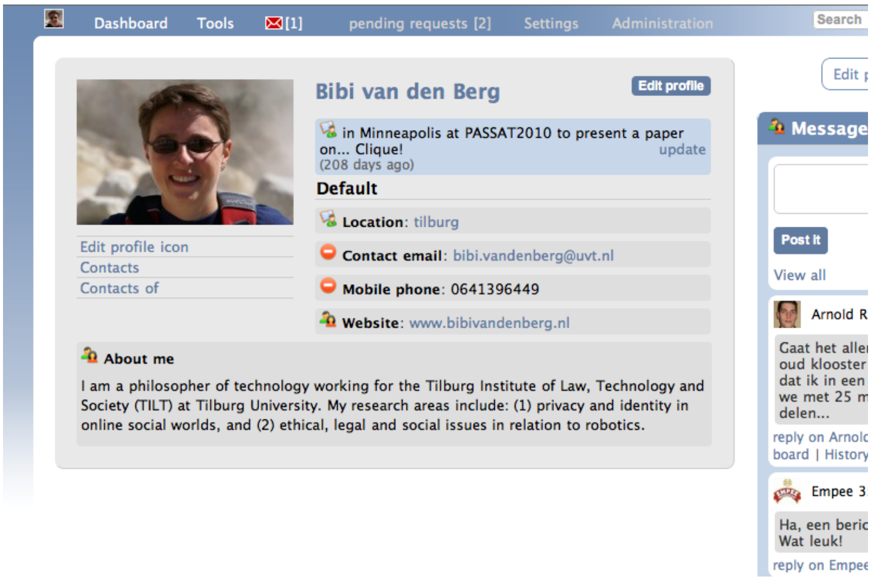
Visibility Rights. The second mechanism we've implemented in Clique to realise audience segregation relates to the *contextualisation of users' profiles* and all of the information published there [Db04]. This entails that information is made public for a specific audience, which may be made up of one or more collections, and/or one or more separate individuals. In Clique, contextualising content and information was implemented by means of two tools. The first is the use of *visibility rights*, which enables users to assign access rights to different collections and individuals. Each time users post items of information or content, they can choose for which audience (both collections and individuals) this item will be visible. For example, a user may decide to make her holiday pictures invisible to her colleagues, but visible to her relatives and some members of her collection of friends, or she may decide to prevent acquaintances from reading her diary entries, but leave them visible to everyone else in her contacts list.

In Clique, individual users can control visibility settings of each individual item of information for two reasons. First, individuals use social network sites to present content with different goals and purposes in mind. Some may use these sites, for instance, only to stay in touch with people they know intimately in the offline world, whereas others may want to use them especially to present (aspects of) themselves before an audience of strangers. Obviously, users thus have different requirements regarding the visibility of their information. Therefore, it would be patronising and limiting if the platform provider would decide for users which information to share and for which (limited or unlimited) audience. Second, users' ideas of which kinds of information are deemed 'private' vary:

> "Different people have different views of what should be private. [...] People must be able to reach their own decisions about what should be private, and what gains they would hope to make by releasing information about themselves" [OS08].

An objection to providing extensive control over visibility settings could be that users don't *want* too much control over their content in social network sites. In fact, researchers have argued that users are not interested in fine-grained control over the display of personal data, for instance because making the profile invisible makes it harder for other people to find them [db08b], or because they would simply find it too much hassle. However, recent research has shown that, when given the opportunity, many people do in fact want to shield some of their information [YQH09], especially since quite a significant number of negative examples regarding information spill and privacy issues with respect to social network sites have been published in the press in recent times.

Clique implements a fine-grained mechanism for setting access control policies, in which each element of the profile can be made visible for either collections, or individuals, or a mixture of both. This means, for instance, that a user can make her name and date of birth visible to everyone, while restricting access to her address to colleagues, and allowing only some designated contacts to see her mobile phone number. Figure 2 shows a user profile page in Clique. Each item contains an icon that displays its current audience on mouse over. Users can choose between the following access control options for the content published on their profile: 'only visible to me', 'contacts/collections' (in this figure the field called 'website'), 'all contacts', and 'public' (e.g., location). When users publish information in Clique they are presented with an access control dialog as shown in Figure 3. In this dialogue window we 'nudge' [TS08] the user to act in a privacy-savvy manner without undermining sociality. By default, the user's primary audience (her default collection) is selected as having access to the content to be published. The user can drag collections and individual contacts to the red and green boxes to grow or shrink the audience. Note that in this case, the users' friends and best friends have


Dashboard Tools [1] pending requests [2] Settings Administration Search

Bibi van den Berg
Edit profile

in Minneapolis at PASSAT2010 to present a paper on... Clique!
update
(208 days ago)

Default

Location: tilburg

Contact email: bibi.vandenberg@uvt.nl

Mobile phone: 0641396449

Website: www.bibivandenberg.nl

Edit profile icon
Contacts
Contacts of

About me

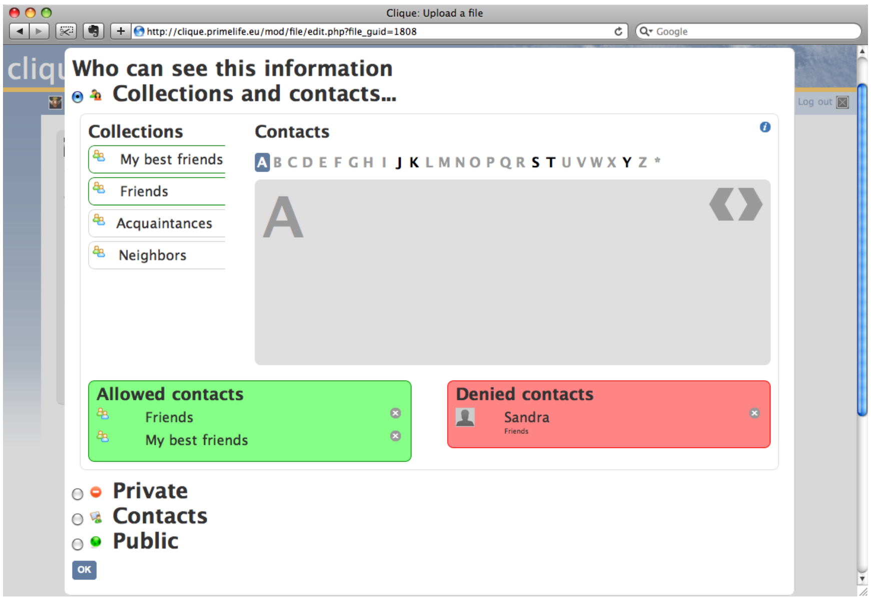
I am a philosopher of technology working for the Tilburg Institute of Law, Technology and Society (TILT) at Tilburg University. My research areas include: (1) privacy and identity in online social worlds, and (2) ethical, legal and social issues in relation to robotics.

Edit
Message
Post it
View all

Arnold R
Gaat het alle
oud klooster
dat ik in een
we met 25 m
delen...
reply on Arnol
board | History

Empee 3
Ha, een beric
Wat leuk!
reply on Empee

Figure 2: Visibility settings in Clique.


Clique: Upload a file
http://clique.primelife.eu/mod/file/edit.php?file_guid=1808
Google
cliq
Log out

Who can see this information

● Collections and contacts...

Collections
- My best friends
- Friends
- Acquaintances
- Neighbors

Contacts
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z *

A

Allowed contacts
- Friends
- My best friends

Denied contacts
- Sandra
Friends

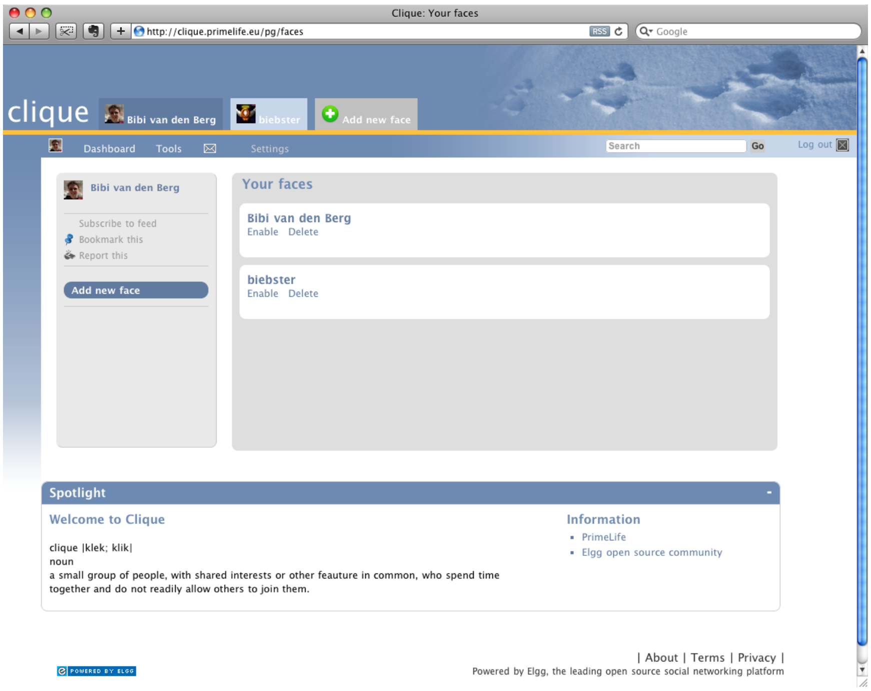
○ Private
○ Contacts
○ Public

OK

Figure 3: Access control dialogue in Clique

access to the content to be published, with the exception of one contact, called Sandra, from the collection called 'friends'. While enabling access to a collection, thus, the user can still choose to make information unavailable for particular individuals.

Faces. The third and final mechanism we have developed for Clique is the introduction of *tabs* to represent what we call the different 'faces' a user may want to combine within the same platform. This second form of contextualisation mimics the fact that individuals maintain different social spheres in the offline world. Most social network



clique |klek; klik|
noun
a small group of people, with shared interests or other feauture in common, who spend time together and do not readily allow others to join them.

Figure 4: Faces in Clique.

sites implement a single profile for each user. All of a user's contacts see the same information. However, for privacy-purposes it is important to allow users to diversify the information and content they present to different audiences. Moreover, many people now maintain different profiles on different platforms (e.g. Facebook and LinkedIn), which is cumbersome and time-consuming. If these profiles could be combined in a single social network site, users would only have to access that single environment to manage multiple, separate self-presentations.

In Clique, the different 'faces' a person may have in the offline world can be recreated using tabs. Each tab functions as a separate social sphere, representing one aspect of the user's identity. For instance, users may create a tab for their private face and for their professional face. Each of these faces contains its own network of contacts, which can be assigned to the various collections within each tab. Access rights can be defined for collections and contacts with regard to all content presented in this context (i.e. using a specific face in front of a specific collection). Contacts only get access to the information that is made visible for them. This means that a) contacts who only know the individual professionally, for instance, are prevented from acquainting themselves with the user's leisurely profile; and b) within each face, contacts can only access the information that is explicitly made available to them. Figure 4 shows what faces look like in Clique. The tabs to distinguish between different contexts are a visually appealing and easy way for the individual to manage their various profile pages (faces) in Clique. Information added to one of the faces (e.g., the 'biebster' tab) is invisible in all other tabs, and hence it is easy for the user to manage who sees what. Clique can therefore be used as a dashboard for multiple social environments. By simply clicking through the different tabs the user can see what information is accessible there, while the audience indicator icons reveal the current audience. Creating faces is a bit cumbersome, since it means that users need to build a new profile, set the security and privacy settings, and add contacts and content for each individual face. They have to invest energy and time in setting up a new profile. Particularly when users create multiple faces for which the contact list shows a significant overlap we may wonder whether users are willing to make this investment, and whether they may see (enough of) the benefits and advantages of creating separate faces.

After this overview of the core functionality we created for Clique we will now turn to a discussion of the improvements we made to the system in the last year.

2.4.2 Improving Users' Understanding of Privacy Issues in Social Network Sites

As said before, we launched the first version of Clique in March of 2010. We deliberately chose to leave out extensive explanations of the goals of, and mechanisms implemented into, the system in that first version, because we were curious to see how users would respond to the system without us 'priming' their responses and behaviours in any significant way, and because we were curious to see whether they would grasp the goals and mechanisms of audience segregation in an intuitive way. We soon realised, however, that some guidance was required, since users did not automatically understand our intentions with the demonstrator. While a significant number of Clique's users presumably became a member of the network because they value privacy, at the same time their behaviours after becoming a member revealed that they did not intuitively appreciate our translation of the ideals of privacy protection through audience segregation in this particular network. To improve the network, users' privacy-protection and their experiences in Clique, therefore, we decided to take two steps:

1. We conducted a large survey in which we aimed to uncover users' needs for, and ideas on, audience segregation in social network sites;
2. We developed two wizards to improve users' understanding of the idea(l)s that drive Clique.

We will discuss the former in the next section, and then describe the wizards we created to help users grasp the meaning and safe use of Clique in the section after that.

2.4.3 A Survey on Privacy Perception and Audience Segregation in a Dutch Social Network Site: Outline, Preliminary Results and Lessons Learnt for Clique

In the winter of 2010 we conducted a large survey among users of social network sites, to find an answer to the following question: *To what extent does the concept of audience segregation from the offline world meet the needs of users of online social network sites?* To answer this question, we posed the following subquestions:

1. To what extent does online and offline privacy-related behaviour reveal audience segregation?

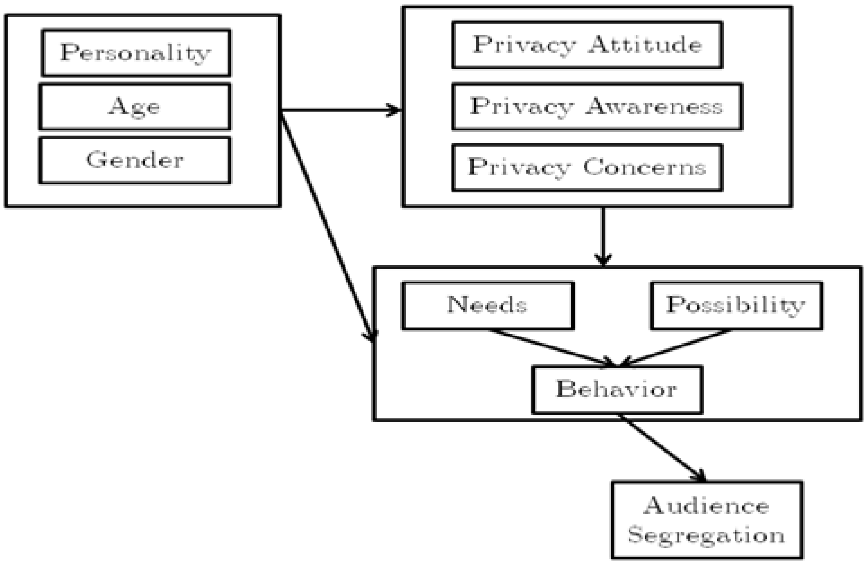


Figure 5: Research model for the survey

2. Do people desire/need a virtual version of a concept such as ‘audience segregation’?
3. Do factors such as age, gender and personality influence the use of audience segregation?
4. Are there differences between online and offline privacy perceptions?

Tim Holweg conducted the empirical study to answer these questions. He used an online survey to collect data. This survey was developed using LimeSurvey[9], a free and open source application. Tim developed the survey in a stepwise manner, based on the research model presented in Figure 5.

Questions about privacy and social network sites characteristics were based partly on the surveys of Acquisti and Gross [AG06] and Fogel and Nehmand [FN09]. The short revised version of the Eysenck Personality Questionnaire (EPQ-rss) was used for measuring introversion/extroversion personality characteristics. The survey contained 41 questions (such as Likert-style scale, dichotomous (yes or no), and open and closed questions), divided into four major sections:

1. Demographics (external factors in the research model),
2. Social network site characteristics,
3. Privacy, and
4. Behaviour.

When designing the survey, questions were ordered in such a way that (answers to) previous questions could not influence the answers to subsequent questions. For example, (answers to) questions about privacy might influence the answers a respondent gives to

[9] http://www.limesurvey.org/

questions about audience segregation. Therefore, questions about privacy were placed *after* questions about audience segregation. Subsequently, the survey was tested by 12 people different in age and gender categories. Test-participants were asked to give feedback about the survey with regard to easiness, understandability and time to complete the survey. An average time of 15 minutes was needed to complete the survey.

After this test phase the actual survey was conducted. A total number of 1163 people participated in the survey. Of the 1163 surveys, 906 (77,9%) surveys were usable. Participants to the survey were recruited in three ways:

1. Through an advertisement on the Dutch social network site Hyves. This advertisement enabled us to announce the survey to Hyves users;

2. Through a request that was sent to 44 different Dutch universities and colleges of higher education. In this request, we asked the board of directors to forward an email invitation with a link to the survey to all of their students; and

3. By sharing the survey via Hyves, Facebook, Twitter and LinkedIn contacts and groups.

To increase the number of participants, and to show appreciation for respondents' willingness to contribute, we offered 2 Ipod nanos in a lottery among all participants. For this study, participation was limited to Dutch-speaking subjects. Participants were asked to complete questions about privacy and audience segregation topics anonymously. Data were collected from December 2010 until 14 February 2011.

In the survey we began by asking users why they make use of social networks in the first place. Suggested answers included: "to meet new people", "to stay in touch with people I already know", "to see other people's pictures and share my own", "to meet people who share my interests", and "to show people which things are important to me". Users could value these answers on a five-point Likert scale, ranging from "very unimportant" to "very important". After that we asked questions on information disclosure. For example, we asked users why (for which reasons) they share information with others. The same Likert-scale was used as before. Suggested reasons included: "I enjoy sharing the things I experience with other people", "I want to improve my social skills (e.g. by participating in a group on a social network site or in a discussion on a forum)", "I would like to become more popular (e.g. by gathering more friends or followers)", "I enjoy spreading news and/or gossip", and "I would like to find a job".

Then we turned to questions on the audience that has accessed to the users' information in social network sites. We asked, for example, who has access to (parts of) the users' profile page, for example to their contact list, the pictures they post online, their contact details, their wall posts, and their interests/hobbies/favourites. Users could choose "everybody", "friends", "friends of friends", "only me", "specific contacts/groups", "I don't know", and "not applicable" to answer each of these elements.

Next, we attempted to uncover users' attitudes towards audience segregation in the real world. We presented them with statements such as "when you get money at an ATM or pay with a bank card, do you ensure that nobody can see your PIN number?", and "if you have an argument with someone over the phone, would you attempt to resolve it while in a public place, such as on a train?" Answers came on a five-point Likert scale again, ranging from "never" to "always".

Finally, we asked after users' attitudes towards audience segregation in the online world. For example we asked them whether they ever wondered who their audience was when posting information online, or when leaving an email address or a phone number in an online environment. The same five-point Likert scale was used as with the previous question.

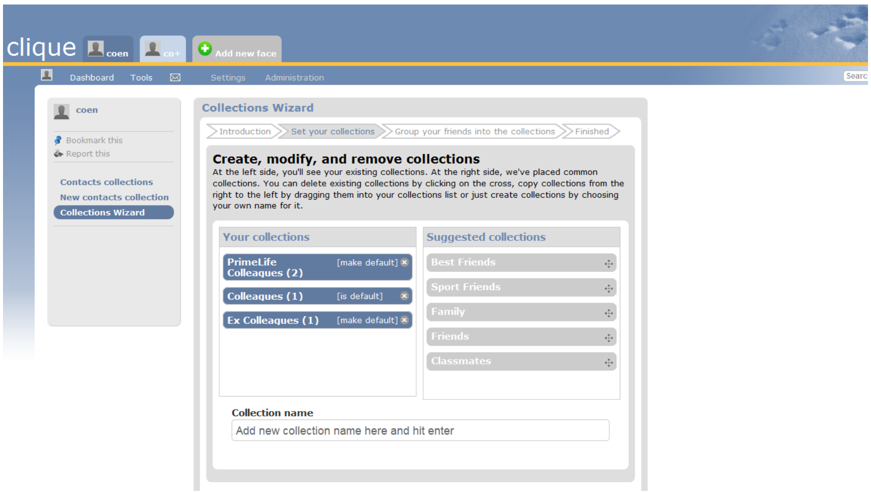
The survey closed on 14 February 2011. We are currently analysing the results and therefore, unfortunately, can only provide a few preliminary details in this deliverable. A few salient details stand out in our earliest findings. Most importantly, these are:

1. When asked how concerned users are about their privacy, the study reveals that most users are not very concerned about privacy in general, but in some specific cases they are very concerned with privacy violations indeed. Specifically, users strongly object to others reading their personal messages (41,8% do mind and 41,2% do mind very much), or others intercepting their personal messages or emails (43,6% do mind and 35,7% do mind very much).[10]

2. When asked about their behaviour in offline and online contexts, and the relevance of audience segregation for these two respective contexts, the survey reveals that audience segregation is indeed an oft-used mechanisms, both offline and online. For example, the vast majority of participants states that they would never (40,2%) or hardly ever (42,1%) discuss a conflict over the phone in a public place, such as a train compartment, since a conflict is commonly a personal matter, while the train is a public place. This reveals that participants are very hesitant to discuss personal matters with everyone, especially people they do not know. Moreover, the answers to other questions such as 'Do you hide your bank card PIN number when using cash machines/making purchases?' (55,3%, always hide their pin number) or 'Would you leave a confidential letter unattended at a school/university or office?' (71,4%, never leave confidential information unattended) confirm that participants know the value of certain information and are aware of their audience. Interestingly, the survey revealed that participants not only use audience segregation in offline worlds, but also in online worlds. Especially when they publish "*high risk*" personal information online, such as a phone number or an e-mail address, users tend to engage in audience segregation. 41,5% of the participants answered that they always keep in mind who can have access to the information when sharing such personal information.

3. Finally, on the issue of audience segregation in social network sites we obtained the following results. 70,8% (n=906) of the respondents answered affirmatively to the question 'Would you like to control access to your online personal information, so that some people can or cannot see (certain parts of) your profile?". Interestingly, at the same time more than 92,3% of the respondents are content with the current technical possibilities to protect their online personal information. Only 18,4% (n=709) of the respondents use the current functionality to create groups of contacts within existing social network sites such as Hyves and Facebook, while 57% knows that this functionality is available.

[10] Note, that these results simply reflect data from two different questions, which we asked to see whether users were consistent.



Collections Wizard

Create, modify, and remove collections

At the left side, you'll see your existing collections. At the right side, we've placed common collections. You can delete existing collections by clicking on the cross, copy collections from the right to the left by dragging them into your collections list or just create collections by choosing your own name for it.

Collection name

Add new collection name here and hit enter

Figure 6: The collections wizard

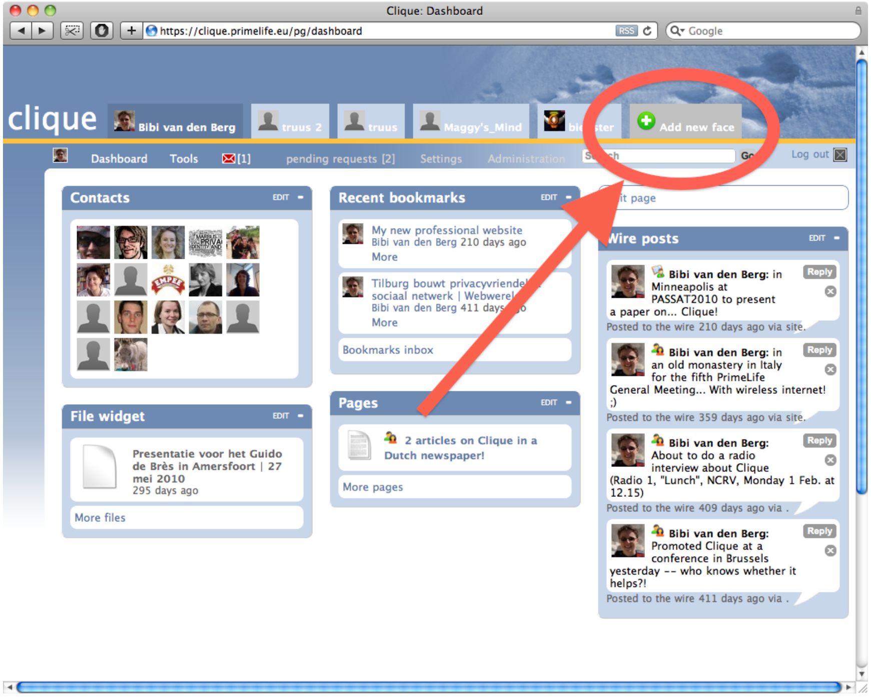
2.4.4 Helping Users to Understand Clique: Two Wizards

As we said at the beginning of this section, after Clique's initial introduction we soon realised users needed some guidance in the use and meaning of the audience segregation mechanisms we had implemented in this social network site. To solve this, we developed two wizards to guide users in the right direction and explain our intentions with Clique's privacy-enhancing mechanisms.

The collections wizard. The first wizard we created was a '*collections wizard*'. This wizard facilitates the creation and maintenance of collections in Clique.

When a new user registers for Clique, the collection wizard follows immediately after the completion of the registration process. The user is then redirected to the collection wizard page, where she is obligated to create at least one collection. In the first of four steps, the user is informed about the concept of collections. We explain what a collection is, and what purpose it serves in terms of privacy protection. In the second step of the process, the user is instructed to create her collections. Creating collections can be done by simply typing a name for the collection or by dragging suggested collections to the container of user-made collections. This is visible in Figure 6. Step 3 of the process is to add existing contacts to collections. After registration, the user will not have any contacts and therefore, this step seems unnecessary. The reason for this step is to make users aware that collections are meant to store contacts in. Since the wizard can also be started when the user already has collections and contacts this step is essential. The fourth and final step is only to inform the user that she has successfully finished the wizard and that she can find the wizard in the menu if she wants to start it again.

Since we wanted to explain the principles behind the creation of collections to both new and existing members of Clique, we implemented this wizard in such a way that (1) new members were automatically guided through the wizard after registration, as explained above, and (2) after its launch, any existing members that did not make use of the functionality of collections yet were shown this wizard the next time they logged



My new professional website
Bibi van den Berg 210 days ago
More

Tilburg bouwt privacyvriende sociaal netwerk | Webwerel
Bibi van den Berg 411 days o
More

Bibi van den Berg: in Minneapolis at PASSAT2010 to present a paper on... Clique!
Posted to the wire 210 days ago via site.

Bibi van den Berg: in an old monastery in Italy for the fifth PrimeLife General Meeting... With wireless internet! ;)
Posted to the wire 359 days ago via site.

Bibi van den Berg: About to do a radio interview about Clique (Radio 1, "Lunch", NCRV, Monday 1 Feb. at 12.15)
Posted to the wire 409 days ago via .

Bibi van den Berg: Promoted Clique at a conference in Brussels yesterday -- who knows whether it helps?!
Posted to the wire 411 days ago via .

Presentatie voor het Guido de Brès in Amersfoort | 27 mei 2010
295 days ago

2 articles on Clique in a Dutch newspaper!

Figure 7: Creating a new face in Clique

onto the system. This way, existing members, too, were pointed towards the meaning and functionality of this aspect of Clique's privacy-enhancement. Currently, many, but not all of Clique's users are using the collection functionality.

The faces wizard. The second wizard we created was a '*face wizard*' to explain the ideas behind and meaning of the notion of 'faces', which we discussed above. In Clique, users can create multiple faces to show different sides of themselves (e.g. personal or professional) in the different faces. Because the concept of faces also required more explanation, and because it turned out that users needed some guidance in the creation of multiple profiles, each with a different template, we implemented just that: a wizard leading to the creation of various profile templates.

The faces wizard works in very similar to the 'collections wizard'. This was designed purposefully to create user recognition and facilitate usability. When a user clicks the tab labeled 'add new face' at the top of the Clique window (see Figure 7), the faces wizard will start up automatically. This wizard also consists of four steps. In the first step we explain what the concept of faces means, and what purpose it serves in Clique. The second step is the most important one. In this step, the user chooses a *type* of face. Four predefined faces are offered, namely: 'personal', 'professional', 'hobby', and 'avatar'. When the user chooses one of these four types, a face will be created with profile fields that match the requirements of that face. For example, in a 'professional' face a user may wish to share her educational background and information about past and current employment, but this information is irrelevant on a profile page about a user's avatar. Similarly, on an avatar profile page the user may wish to share information about the nickname of the avatar, the game or virtual world in which it is active, and the achievements this avatar has accomplished (e.g., the game level it has reached), but she may not necessarily want to share personal information such as a (real) name or telephone number in this profile page. To accommodate these different profile type requirements, Clique's administrators have created the aforementioned four *profile templates*, each containing specific fields.

As said, in the second step of the wizard, the user chooses which type of template she wishes to use. If the user does not want to use one of the four predefined templates, there is an alternative option, labeled 'Custom'. If the uses chooses this option, no predefined fields will be offered in the profile template, except for a screenname and e-mail address, so that the user is as free as possible to taylor the profile page from scratch. To safeguard usability, when users choose this option a message is shown to the effect that no predefined fields will be included in the profile page, so that users are aware that they need to do more work on the page if they decide to go for this option, rather than one of the predefined templates. Note that, once a template is chosen, users still have freedom to add, remove, or change fields and field categories.

In the third step of the wizard, the user chooses a screenname for this face. This step comes *after* choosing the profile type, since the screenname may depend on the choice made in step 2. After all, if a user has chosen to create a face for her Second Life avatar, for example, she may wish to use a different screenname from the one she would have used, had she chosen to create a professional face. In the final step the user is informed that the face is now completed, and asked whether or not she would like to continue by creating collections for this face. If so, the collections wizard will be activated immediately after this step. The faces wizard has been launched only very recently. Most users in Clique still have only one face, but we hope that with the implementation of this wizard the number of faces will go up considerably.

2.4.5 Solving Privacy Problems Caused by Tagging in Clique

One of the most important features of many web 2.0 environments is that of *tagging.* In computer systems terminology, a tag is a *keyword* or *term* assigned to a piece of information or a resource (for example to an image or a document). This kind of metadata helps to describe the resource and allows it to be found again by browsing or searching. Tags are generally chosen informally and personally by the resource owner or by its viewer, depending on the system. Tagging (sometimes called also 'labeling') is carried out to perform functions such as marking ownership, and indicating online identity. It may take the form of words, images, or other identifying marks.

While tagging functions as an easy and efficient way of adding metadata to content, at the same time this mechanism may cause privacy problems. One area where this is especially relevant is in social network sites and other forms of social software. When tags consist of individuals' names, for instance accompanying a picture in which they are displayed, this may be a privacy violation for the individuals involved. Especially when individuals are tagged on a resource that doesn't belong to them, the privacy problem is all the more poignant, because the individual has limited means to undertake countermeasures. Damage to users' image through tagging can have consequences in the real world, too. This is why we concluded that it is important to provide social network site users with instruments to protect their privacy against (unwanted) tagging in Clique. We decided to build an extension for Clique (and Elgg) to support privacy-oriented tagging.

During the design phase, a preliminary step was to define precisely what it means to tag a user on a resource, and how this is usually accomplished in a social network site. To answer this question we analysed the tagging practices in the most commonly used social network sites, and we developed the following definition:

> A user is tagged on a resource when her social network nickname, or her account name, is used directly during the creation of the resource (e.g., if the resource is a piece of text, such as a note) or if her nickname (or account name) is used subsequently to label an already created resource.

After we had set this definition, we designed a model representing our tagging system and describing the actions a user can undertake in this system. The actors of this model are (1) the social network *users*, which can be subdivided into four overlapping classes, and (2) the *resources.* The classification of the users is represented in as follows:

1. The resource *owners*: the users that upload or create the resource in the social network site. This kind of user can define the access rights for their resources using the tools of the social network site;

2. The *tagging users*: the users that tag others user in a shared resource which they have access to;

3. The *tagged users*: the users that have been tagged on a resource;

4. The *contacts* of the *tagging user*: the users belong to the tagging users' contact lists.

When a tagging user tags another user on a resource, the system tries to match her contacts with the labels used. For any positive match the system creates a *privacy notification request* that defines which user has been tagged by whom on which resource.

> A **privacy notification request** is an expression of the form ⟨ *subject* ⟩ TAGS ⟨ *contact* ⟩ ON ⟨ *resource* ⟩ BELONGS TO ⟨ user ⟩.

Where ⟨ *subject* ⟩ belongs to **tagging users**, and ⟨ *contact* ⟩ belongs to **contacts** of the **tagging user**. When a notification has been created, the resource can be accessed only by its owner and by the tagged users, until all the tagged users accept to grant access to the resource itself. When a tagged user takes a decision about a resource (i.e., grant or deny access to the resource itself) the privacy notification becomes a *privacy policy* enforced by our extension.

> A **privacy policy** is an expression of the form ⟨ *subject* ⟩ GRANTS ACCESS | DENIES ACCESS ⟨ *resource* ⟩ BELONGS TO ⟨ user ⟩.

Where ⟨ *subject* ⟩ belongs to **tagged users**. It is important to note that the access control management of a resource can be represented as a two-layer access control system: the first layer is the standard social networking access control system that enforces

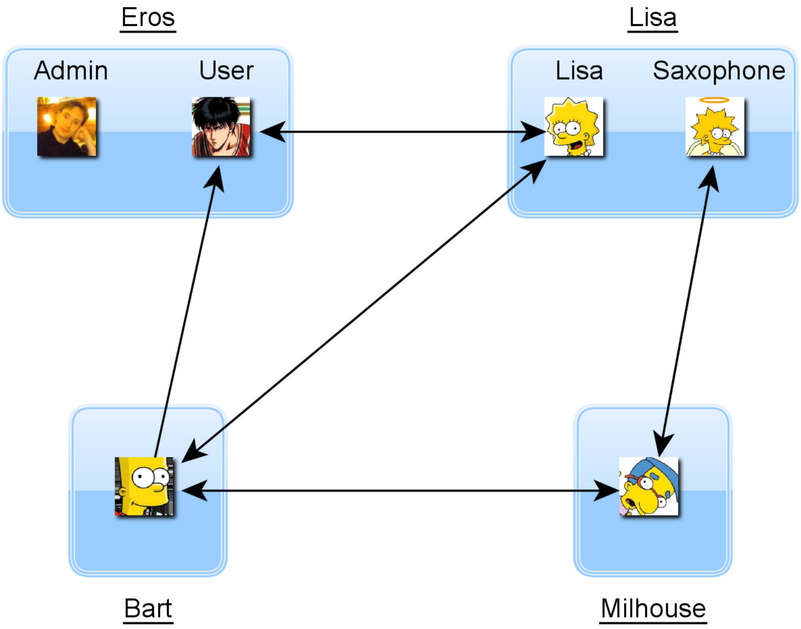


Figure 8: Graph representing friendship relations among the users.

the policies defined by the resources owner, while the second layer is the access control defined in this extension that takes in account the decisions of each tagged users within a resource. If at least one of the tagged users *denies access* to the resource, the resource itself can not be published. In this way it is possible to limit access to a resource where a tagged user feels her privacy violated.

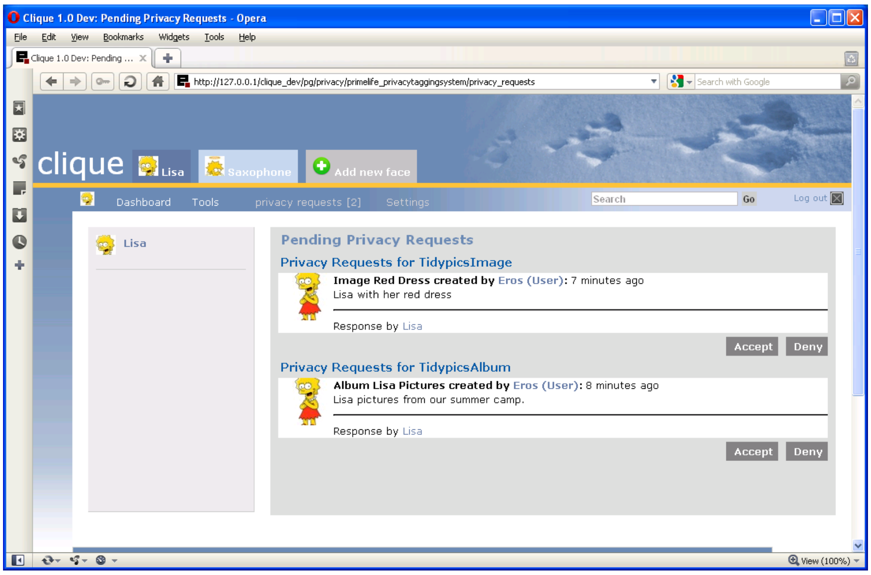
In our model, we take into account the possibility that an already tagged user is un-tagged. This kind of situation can be managed in two ways:

1. The privacy notification (or the privacy policy) related to the un-tagged user has to be removed;

2. Even if the user becomes un-tagged her privacy notification (or her privacy policy) has to be enforced in any case.

We chose the second approach for two fundamental reasons. The first one revolves around *consistency*: a tagged user knows the existence of the resource and the link with her, so from our point of view it is correct that the user can continue to have decision power over the resource. The second reason is to avoid unethical behaviours such as would occur when, for example, the tagged user denies access to the resource, so the resource owner or the the tagging user un-tags her.

The Privacy Tagging System in Practice. To show how the privacy-enhanced tagging extension that we built for Clique works, we will describe a step-by-step example. In this example, let's assume that Clique has four registered users:


Clique 1.0 Dev: Pending Privacy Requests - Opera
File Edit View Bookmarks Widgets Tools Help
Clique 1.0 Dev: Pending ...
http://127.0.0.1/clique_dev/pg/privacy/primelife_privacytaggingsystem/privacy_requests
Search with Google
clique
Lisa
Saxophone
Add new face
Dashboard
Tools
privacy requests [2]
Settings
Search
Go
Log out
Lisa
View (100%)

Pending Privacy Requests

Privacy Requests for TidypicsImage

Image Red Dress created by Eros (User)**:** 7 minutes ago
Lisa with her red dress

Response by Lisa

Privacy Requests for TidypicsAlbum

Album Lisa Pictures created by Eros (User)**:** 8 minutes ago
Lisa pictures from our summer camp.

Response by Lisa

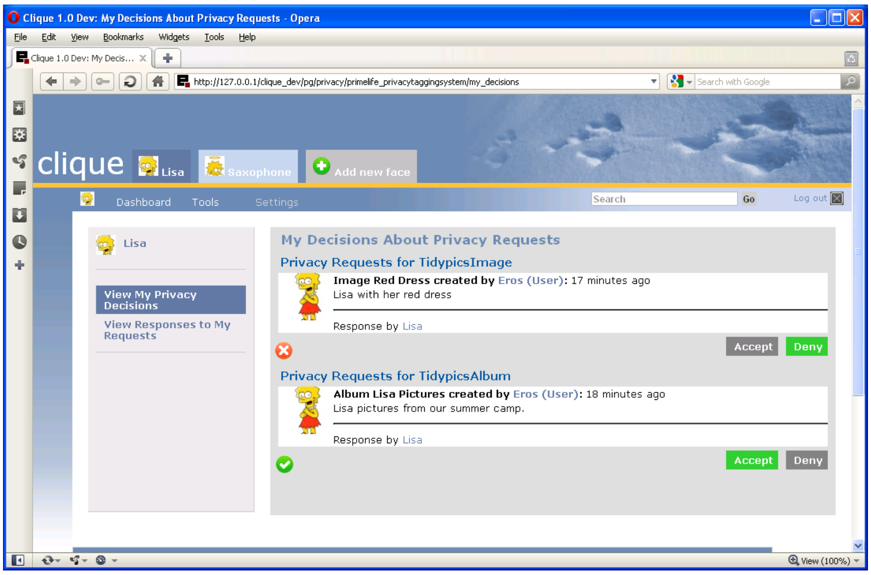
Figure 9: Privacy requests for Lisa: Management page

- **Eros**: he has two faces: *Eros(Admin)* (the administrator of the social network) and *Eros(User)* (a simple user);
- **Bart**;
- **Lisa**: she has two faces, too: *Lisa* and *Saxophone*;
- **Milhouse**.

The following relationships exist between these four users (cf. Figure 8):

- Eros(User) is a member of the contact list of Lisa and Bart, while Eros(Admin) is in nobody's contact list;
- Lisa is a member of the contact list of Eros(User) and Bart;
- Bart is a member of the contact list of Lisa and Milhouse;
- Milhouse is a member of the contact list of Bart and Saxophone; and
- Saxophone is a member of Milhouse's contact list.

Now, let's say that Eros(User) wants to create a new *public* album containing pictures of Lisa at this year's summer camp. The tagging system will then check whether any of the name 'Lisa' appears in Eros(User)'s contact list, and if this is the case – which it is – a privacy notification will be created in order to inform Lisa that Eros(User) has created an album to which her name is attached as a tag. Similarly, each time Eros(User) upload a picture to the album, to which he adds the tag 'Lisa', the tagging system will check his


Clique 1.0 Dev: My Decisions About Privacy Requests - Opera
File Edit View Bookmarks Widgets Tools Help
Clique 1.0 Dev: My Decis...
http://127.0.0.1/clique_dev/pg/privacy/primelife_privacytaggingsystem/my_decisions
Search with Google
clique
Lisa
Saxophone
Add new face
Dashboard
Tools
Settings
Search
Go
Log out
Lisa
View My Privacy Decisions
View Responses to My Requests
Accept
Deny
Accept
Deny
View (100%)

My Decisions About Privacy Requests

Privacy Requests for TidypicsImage

Image Red Dress created by Eros (User): 17 minutes ago
Lisa with her red dress

Response by Lisa

Privacy Requests for TidypicsAlbum

Album Lisa Pictures created by Eros (User): 18 minutes ago
Lisa pictures from our summer camp.

Response by Lisa

Figure 10: Managing the privacy policies page in Clique

contact list to notify Lisa that she has been tagged.[11] Figure 9 shows the notification that Lisa receives after Eros(User) has tagged an album and a picture with her name.

Lisa can then decide either to grant or to deny access to the album and to the picture. Let's imagine that Lisa decides to grant access to the album, but to deny access to her picture. To do so, she clicks the *Accept* button for the album, and the *Deny* button for the picture. Now the tagging system transforms the privacy notifications into privacy policies.

If, in the future, Lisa changes her mind, she can change her privacy policy clicking on the *Setting* item on the top bar, and then selecting the item *View My Privacy Decision* in the right menu. Figure 10 shows the current status of *Lisa* decisions. The central feature of this privacy-enhancing extension in Clique is, then, that tagged content will only be made visible *if, and only if* the individuals who got tagged have consented to their publication. As long as these individuals do not consent, or deny their publication, the content will not be visible to others.

2.4.6 Decentralised Social Networking for Clique

With the creation of Clique, we initially aimed to build a privacy-enhanced social network site for *individual users*, i.e. we focused on creating mechanisms that enable users to better protect their privacy when engaging in interactions with others, using our site. To that goal, Clique provides users with features to control who has access to content posted in the network. In a word, Clique provides preservation of privacy for its users on a *personal level* by applying the concept of audience segregation into the structure of a social network site.

[11] Note that in the demonstrator the tag is replaced by a link to the searching functionality of clique. So by clicking on a tagged user, one will be able to find the user's profile and other resources related (via the tag) to the user as well, but only if you have the right to access them.

However, only tackling the privacy issues in social network sites on the *personal* level does not guarantee a solution for *all* of the privacy issues that may arise in social network sites. As we have argued in an earlier heartbeat [WP109a], privacy issues in social network sites may also be caused by the service providers of these sites or by third parties. For example, if the service provider provides access to the network for third parties through apps that are hooked into the network, these third parties may gain access to users' personal information. Moreover, if service providers do not secure the network properly, information that users post in the network may leak to individuals or businesses outside that network. Or, service providers may themselves be a privacy hazard to users by storing and monitoring all of the content that users post online. While one could argue that service providers have an obligation to patrol the network in order to prevent users from posting illegal material, at the same time their ability to access all information that users post within the network is a privacy threat from the perspective of (innocent) users. These examples reveal that the behaviours of both third parties and service providers need to be examined if a social network site aims to be truly privacy-friendly, and mechanisms need to be deployed to replace their (possibly) privacy-threatening behaviours with more privacy-enabling ones.

In light of these issues, we chose to focus on the issue of service providers' full control of all user data first, to see if we could find a more privacy-friendly alternative in Clique. In current social network sites, the service provider (or platform owner) has access to, and complete control over, all of the content and data posted in the network by users. We proposed to change this practice in Clique by attempting to realise a *decentralised network*, thus improving users' privacy. We aimed to create an extension for Clique that would be central-platform-independent. The goal was to facilitate *distributed user data and content storage*.

In the past few years, a number of efforts have been made to create decentralised social network sites, both in various academic contexts and in online development communities. We have studied and researched a large amount of related literature, and have looked at the various social network sites that have decentralised features already. Roughly, decentralised social network sites can be placed into two categories: (1) distributed social network sites, and (2) P2P (peer-to-peer) social network sites. The former are social network sites that provide decentralisation through federation of data servers controlled by independent parties, while the latter are social network sites that depend mainly on a P2P structure of client nodes in the network.

After this study of the existing landscape of decentralised social network sites, we set to work on creating a new version of such a network for Clique. Our first plan was to develop a *USB-stick-based client-central-server architecture* named *'roaming Clique'* as a decentralised extension of Clique. In this architecture, Clique's users could choose to store their (sensitive) personal data on a portable USB stick, which would function as a decentralised node in the social network. The data that were to be selected to be stored only on the personal USB stick would be available to others if, and only if, the user chooses to insert the USB stick into a machine connected to the Internet. If the user would not insert the USB stick into a machine, the data would be unavailable to others.

One assumption in the above proposal is that there will still be a central Clique server, which is used for communication between users. After all, it is only the individual users' personal data that is stored on the USB stick. To further improve the privacy-friendliness of the network through decentralised social networking, we are looking into developing an architecture for Clique that allows users, i.e. the peers in this social network site, to communicate in a P2P-fashion. Thus, the central Clique server would only provide basic services, such as a look-up service for nodes in the network and providing PHP for the basic elements of the web page. In order to ensure that the central Clique server cannot gain access to the users' data, which will be stored on the client side (the USB stick), cryptographic schemes must be applied. We have looked into the possibilities of applying different kinds of cryptographic schemes in this environment, and based on that research we aim extend the original plan for decentralised social networking – using the USB stick – to include an architecture that uses *attribute-based encryption* and *access control lists* with the help of the browser on local machines. Attribute-based encryption fits in neatly with the concept of "collections" in Clique, and provides possibilities to secure the data towards different groups of audiences.

2.4.7 Other Improvements in Clique

In response to the way Clique developed over the course of its first year online, we constantly adjusted the network and its layout in response to users' behaviours and needs. Some of the (small) improvements we made in this respect are listed here:

1. In the Elgg software users can set visibility of information in a profile page to several options. One of these is labeled 'private'. This means that no one can see the information posted, except for the user herself. Soon after Clique's launch we noticed that the vast majority of users set (almost) all of their information to 'private'. We were quite intrigued by this choice. After all, why put information online in a social network site while not sharing it with anyone? We soon realised there was a language error at the heart of this behaviour. In the earliest days Clique drew predominantly from a Dutch user pool only, because the network received considerable media coverage in the Netherlands. We realised that many of the Dutch, i.e. non-native English, users probably assumed that the English word 'private' meant that their information would be '*privacy protected*', for instance safely stored. They did not realise, that by choosing this option they were sharing no, or hardly any, data at all, thus defeating the purpose of creating a profile page in a social network in the first place. In response to this misunderstanding, we changed the text of this visibility setting to 'only visible to me'. After that, many users changed the visibility settings of their information (thus proving our assumption was correct), and new users only rarely chose this setting.

2. As with any other website, spammers also found their way to Clique. This resulted in fake blog posts polluting the system. To counter this, we developed a plugin using the API provided by Akismet[12]. Akismet is a webservice that can detect spam in, for example, commments. The plugin sends all public blog posts to this service to be checked for spam. If spam is detected, the blog post will be set to 'private', i.e. it will no longer be visible to other users. All detected spam is

[12] http://akismet.com/

displayed in the administration backend, where the admins can delete or restore these messages.

3. In the earliest days of Clique, users pointed out that Clique profiles could be found using the Google search engine. In terms of privacy protection this, of course, was not an ideal situation. To protect users' privacy in an optimal way, their profile pages ought not to be searchable through Google or any other search engine. Rather, it ought to be impossible to find out whether or not an individual is a member of this social network site in any other way than by becoming a member oneself. Hence, we adjusted the settings of the system in such a way that profiles are no longer findable by Google's (or other engine's) crawlers.

2.5 Scramble!: Using Cryptography to Protect Users' Privacy in Message Exchange

The second demonstrator we developed to improve privacy in social software is called '*Scramble!*'. Scramble! is a Firefox plugin that enables users to post and share messages using, for example, a social network server (or any other type of social software), yet at the same time restrict access to the content of the message so that only a limited set of readers can access it. The set of readers is defined by the user herself. We use cryptographic techniques to provide confidentiality of the data and recipient set, and guarantee the integrity of the posted data. We will briefly describe how this works.

2.5.1 Overview and Goals

As said, Scramble! aims to enable users in a social network site (or any other community sharing information using social software) to exchange messages (data) in such a way that unwanted audiences cannot read them. In order to accomplish this, we use cryptographic techniques. In Scramble!, each user holds an OpenPGP key pair, composed of a public key and a secret key. The public key is known to all users in a user's contact list, while the secret key is known only to the user. We assume that users exchange their public keys when a contact connection is established (i.e. when they accept each other on their contact list), using an authenticated offline channel.

Goals. As we have argued earlier in this deliverable, from a privacy perspective it is very important that users of social software can control their own data, and specify who can access these data, preferably without relying on third party servers, such as the social network site providers, or any other third parties. To contribute to this goal, Scramble! has the following goals:

1. *Privacy Preservation*: Users should be able to define the set of recipients that they wish to authorise to read the data they are sharing. In this way, only users from the correct set will be able to read the raw data. Both the composition of the set of recipients and the value (content) of the data themselves ought be kept hidden from the provider. The confidentiality of the data should be achieved using cryptographic techniques. However, once the data is distributed among the users

in a designated set, there is no way to prevent a *malicious* user in that set from copying, saving or re-distributing the raw version of the data. In this case, the receiving user breaks the social contract.

2. *Publisher Integrity*: Scramble! should guarantee data integrity when it is posted in a social network site, using cryptographic techniques. This avoids attackers from tampering with the value of the data and to impersonate other users.

3. *Deployability*: Scramble! is meant to be deployed in the real world. Thus, it must be stable and compatible with different environments, and social network site independent.

4. *Usability*: Scramble! should present a user interface that is easy to use, in order to overcome usability issues presented in, for example, [WT99]. The operations should be simple and the cryptographic techniques transparent. Operations such as the creation, importing and exporting of keys should be effortless. If a user is not authorised to read a given message, then Scramble! should hide this message from that user.

Key Management. As said, Scramble! uses a combination of public and private key pairs. These keys consist of the public and private keys of an ElGamal encryption and a DSA signature scheme. The keys can be either generated (default behaviour) or imported (power-user behaviour) by the user upon Scramble! initialisation. For example, if a user Alice wants to share a message (data) with a set of contacts, she must possess the associated public keys of all the users in that set. All public keys of this set are stored in Alice's machine, and are managed by Scramble!.

Key management is a hard problem due to the possibility of key tampering and the fact that it is counterintuitive to ordinary users. A malicious user or the social network site provider could replace the public key of the user to impersonate her. Thus, it is important that users can correctly distribute their public keys, as they are used for encryption when posting content. However, if users are not able to exchange any keys and resort to unencrypted alternatives, they are even worse off.

Users have to be able to exchange their public keys when a connection between them is established, for example when they agree to be added to each other's contact list. They can make their public key available using the provider or a key server and should verify fingerprints using an offline channel to verify the authenticity of a public key. As Scramble! makes use of the OpenPGP standard, we make use of any public PGP server. We opted to verify the authenticity of keys manually as the current OpenPGP web of trust has proved to be too complicated for ordinary users [WT99]. Users have to either make a leap-of-faith or check the fingerprints. For future versions it should be easy to introduce a web-of-trust mechanism, if this is desired by power-users.

Alternatively, our key management model could be extended by making public keys available over a social network site-based mechanism, such as the one proposed by [BMP$^+$09], where users cross-certify their digital certificates using social network site relationship connections. The cross-certification is achieved by users signing other users' digital certificates, which are composed by the public key together with some Personal Identifiable Information (PII).

For key revocation or key update users are required to distribute a new public key. However, this only affects new content.

Access Control Policies. We consider that the relationships that each user has in a social network site are represented in Scramble! by the public keys of the users to which she is connected. Moreover, a user can define collections that function as subsets of the complete list of collections (s)he has, in order to separate that list into categories.

Whenever a user publishes a new document or message in the social network site (s)he can define with whom to share it. To do so, the user selects a subset from her contact list, which is to be authorised to read the data. This subset may be composed of single users, of one or more pre-defined collections, or of a mix of both. The subset can be different for each item of content posted. If the user chooses to update an earlier posting, (s)he must re-posted the subset as well.

Cryptographic primitives. For the confidentiality and integrity of data posted in social software, such as social network sites, we had to choose between traditional hybrid-encryption techniques, such as OpenPGP [Zim95], or broadcast encryption such as presented in [BBW06, BGW05]. In both cases, whenever users wanted to share a message with a particular set of contacts, their public keys would be used to create the access list that would be attached to the content to be posted. The confidentiality of the data is then achieved using an encryption algorithm, while the integrity thereof is assured by signing the data before encryption.

We chose to use OpenPGP, since it is a well-deployed standard with support for multiple recipients encryption using hybrid encryption. Moreover, most broadcast encryption schemes such as those cited in [BGW05] do not provide key privacy[13], with the exception of the scheme presented in [BBW06]. The latter, however, also uses a hybrid-encryption approach internally and does not offer performance advantages.

Thus, a message to be posted is encrypted with a one time random-generated secret key, using a symmetric algorithm. Then, for the set of proposed recipients of the message, encryptions of this one time random-generated key are created using the public key of each subject in the set. The integrity of the data is assured by signing it before encryption.

The public key encrypted values of the one time random generated key are appended to the symmetric encryption and represent an anonymous version of the data for the set, which specifies which other users are allowed to see the content. While this will increase the storage overhead on the server side, at the same time it will save the user from managing a large number of different keys for every new item of content (messages, documents etc) on her machine. In addition, it allows the user to keep her defined access sets anonymised, and enforce different access control for single values of the same data. It is important to note that OpenPGP uses an ElGamal encryption key and DSA signing key to perform the previous operations.

In order to keep the set of recipients hidden, we use the *hidden-recipient* option. This option conceals the key IDs of recipients in the encrypted content. In this way,

[13] Here “key privacy” means that an attacker is not able to decide which (public) key was used to produce a given ciphertext.

only the users in designated set are able to retrieve the value of the data. Other users, and the social network site provider stay oblivious of the raw value of the data, learning only that data is being exchanged.

2.5.2 The User Interaction Flow in Scramble!

Scramble! consists of two modules. The first and main element, Scramble!, consists of a Firefox extension that contains the cryptographic primitives to enforce the access rights, and the key and group management. The second and optional element is a TinyLink server. This server simply facilitates content posts and returns a link to the location of the content. We assume that users can choose their external server or set their own server with our provided implementation. We will describe the two elements using the flow of operations needed to publish and retrieve data on a social network site. The process flow is preceded by an initialisation phase.

Initialisation. Imagine that Alice is a member of our social network site called Clique and she does not want to disclose more information than the necessary with the provider. She has 12 contacts in her contact list and wants to send a message that is visible only to three of those contacts: Bob, Charlie and Dave. To accomplish this without trusting the Clique provider Alice uses Scramble!. First, Alice registers to Scramble! that generates a new key pair (a public and a secret key) based on username (email address) and password. She then uploads the username along with her public key to the key server. Then, she obtains the keys for her contacts based on their email addresses. In order to import her contacts into the Scramble! interface, Alice could, in a future version, extract the contacts from the social network site provider directly, using the mechanism described in [Dan09]. For now, imports need to be done manually based on the email addresses of users. To note, that the key management is hidden from the user by the usage of a username and password.

Posting content. As said, Alice wants to post a message in Clique that ought to be visible only to Bob, Charlie and Dave. In the Scramble! interface she creates a set that contains these three names from her contact list. Then, Scramble! signs the message she wants to send and encrypts it with the keys of the authorised users in the set, Bob, Charlie and Dave. If the social network site limits the length of the posted data, then Scramble! posts the encryption of the original message in the TinyLink server, which will return a tiny link to the stored location. After that, Scramble! posts the encrypted value or the tiny link[14] to the original data in the social network site.

Figure 12 shows Scramble!'s user interface when Alice selects a set of users to which she wishes to disclose a message.

Retrieving content. The decryption of encrypted content from the social network site is transparent to the user, as is shown in Figure 13). First, Scramble! reads the encrypted value of the original message from the social network site. If the content is a tiny link, then Scramble! uses the tiny link retrieves the original message from the

[14] The link is encrypted and points to plain text data.

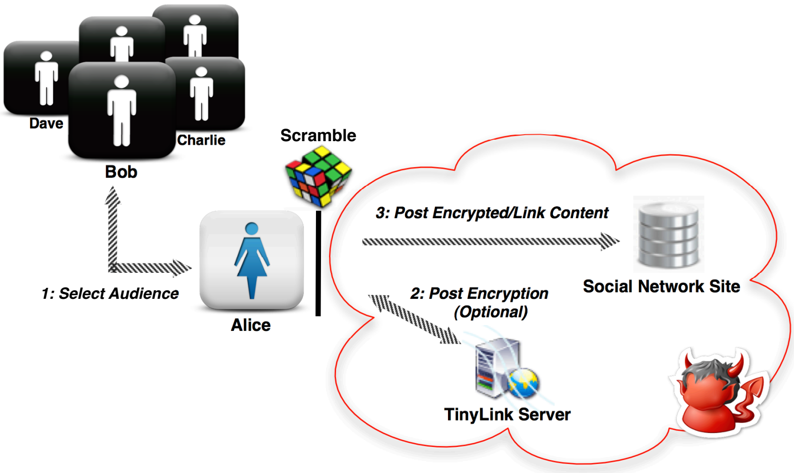
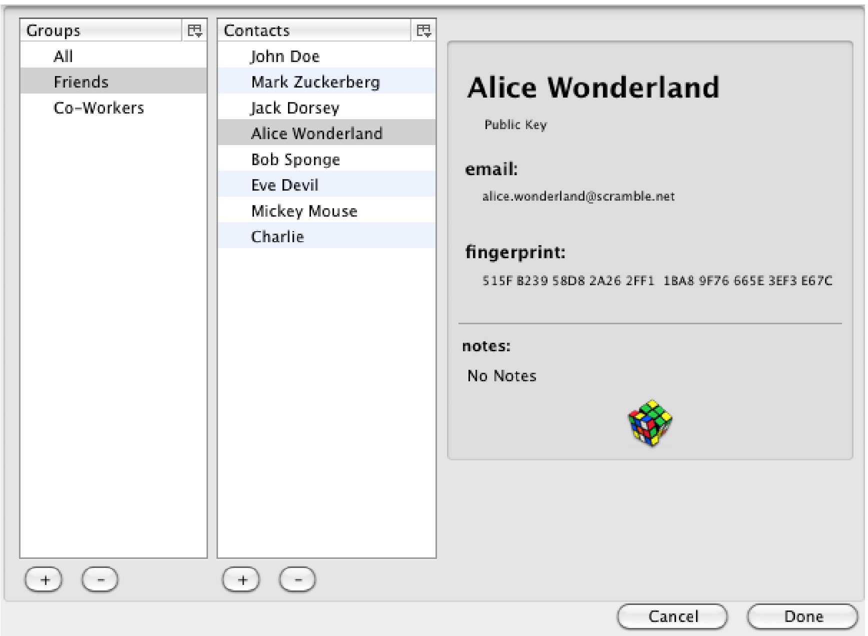


Figure 11: The process of posting new content when using Scramble!



Groups

All
Friends
Co-Workers

Contacts

John Doe
Mark Zuckerberg
Jack Dorsey
Alice Wonderland
Bob Sponge
Eve Devil
Mickey Mouse
Charlie

Alice Wonderland

Public Key

email:
alice.wonderland@scramble.net

fingerprint:
515F B239 58D8 2A26 2FF1 1BA8 9F76 665E 3EF3 E67C

notes:
No Notes

Figure 12: Scramble!'s user interface

TinyLink server. Subsequently, Scramble! tries to decrypt the message and if successful, it verifies if the encrypted version was not tampered with, and that it was in fact Alice who signed the data. Since the data came from Alice and Bob was a member of the intended audience (he is in the set that Alice has created when sending the message), Bob is authorised to read the message. Thus, Scramble! presents the value of the data to Bob. Otherwise, the decryption fails, and the retrieved value is not shown.

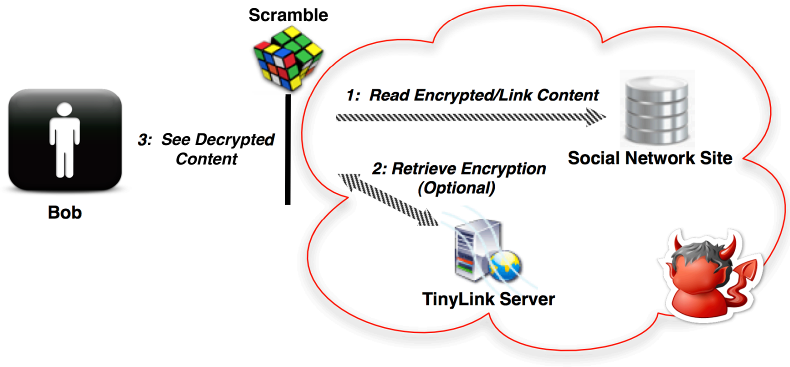


Figure 13: The process of reading content when using Scramble!

2.5.3 Implementation

We have implemented Scramble! as an open source application[15] under the EPL licence [Fou07]. We will now describe the details of the application modules along with their functional aspects. As said above, our implementation is composed of the two modules, a Firefox extension and a Tiny Link Server. We will describe each in turn.

The Firefox extension

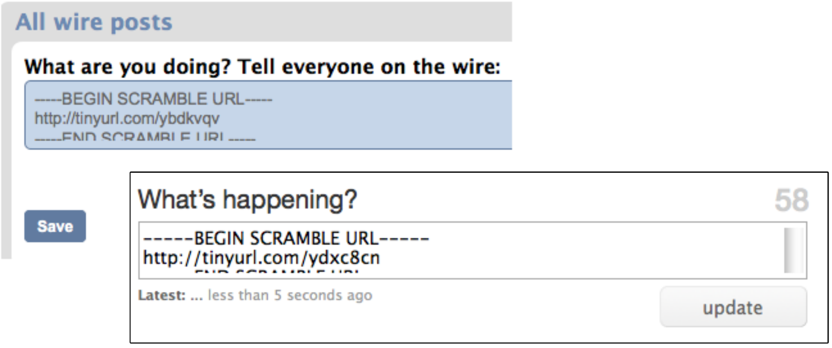
Scramble! is a client-side application implemented as a Firefox extension, which allows cross-platform client-side encryption and key management. Due to the fact that a Firefox extension is developed mainly in JavaScript, we have used a Java XPCOM[16] component to improve performance of the cryptographic module. The Java XPCOM component contains an implementation of the OpenPGP standard. The component executes either a BouncyCastle[17] (BC) OpenPGP implementation or the GnuPG[18] binary module that implements the OpenPGP standard. By means of having the two different implementations, the user can choose to have an embedded OpenPGP implementation with a dedicated key ring with BC, or to execute the general GnuPG module with a key ring that can be shared with other programs.

In order to perform the operations to store and retrieve data from the tiny link server, Scramble executes XMLHttpRequest Post and Get calls in JavaScript. Users that are not using Scramble will see the raw data (as shown in Figure 14). This can be either the full encrypted block or a link to the block.

The tiny link server

The Tiny Link Server was developed to target the limitation of content size imposed by social network site providers. The PHP[19] server stores encrypted data and returns a

[15] A Scramble! version can be found on the project website at http://tinyurl.com/ScrambleIt
[16] http://www.mozilla.org/projects/xpcom/
[17] http://www.bouncycastle.org/
[18] http://www.gnupg.org/
[19] http://php.net/



-----BEGIN SCRAMBLE URL-----
http://tinyurl.com/ybdkvqv
-----END SCRAMBLE URL-----

-----BEGIN SCRAMBLE URL-----
http://tinyurl.com/ydxc8cn

Latest: ... less than 5 seconds ago

Figure 14: Scramble in Twitter

tiny link (a short url) which represents the index of the encryption of the original data. This server can be controlled by the users directly or outsourced into a cloud server, requiring or not extra authentication. We provide the users with the source code and details for their own implementation.

2.5.4 Improvements in Scramble!: Security Analysis, Performance & Usability

After the initial creation of Scramble! demonstrator, we realised that a number of improvements should be made in the final year of the PrimeLife project. Most importantly, we conducted a security analysis, improved performance and conducted usability tests. We will now discuss what each of these improvements entails.

Security analysis

We began by carrying out a Scramble! security analysis, to verify whether the current implementation was protected against various possible attacks. We focused especially on the following elements:

1. *Recipients set anonymity*: Scramble keeps the data that a user wishes to share (e.g. a message or a document) confidential using OpenPGP encryption. In order to anonymise the recipients set for outsiders, Scramble! uses the OpenPGP option *hidden-recipients* to conceal the key IDs. However, this does not offer anonymity of the set of recipients towards a malicious user *within* the set, as shown, for example, in [BBW06]. We note that Scramble! does not provide protection against traffic analysis, meaning that the provider could infer who has access to the content by analysing download and upload operations. Protection against this kind of attacks is left as a subject of future work.

2. *Active Attacks* In an active attack, a malicious server provider attempts to tamper with an item of content that is being shared, by compromising the content's integrity and confidentiality. In Scramble! the user posts the content item in an encrypted format on the server to provide confidentiality of the data. A malicious

server can also have the objective of fooling or impersonating users by changing or replacing the data by its own. In order to prevent such attacks, the user posts the data together with a signature of the data in encrypted format.

Performance

Next we analysed Scramble!'s performance both in light of users' requirements and those of the systems in which it may be used. If individuals are to use Scramble! frequently and easily, its operations must be quickly executed to minimise the overhead of the implementation. Using an XPCOM component we are able to execute a BC Java OpenPGP implementation and the GnuPG module, executing encryption, decryption and signing very efficiently. In our analysis of the performance of our implementation, we focused on the cryptographic algorithms that represent the most expensive operations and the response of the tiny link server, which include the network latency and server process. Scramble!'s performance depends directly on the amount of recipients in a designated set per encrypted block of data. The encryption and decryption costs are represented in Figure 15, where the size of the contact set for encryption and decryption operations goes from 0 to 720 contacts[20]. The public key operations are the most costly operation compared to the use of a symmetric encryption algorithm. The performance complexity details can be described as follows:

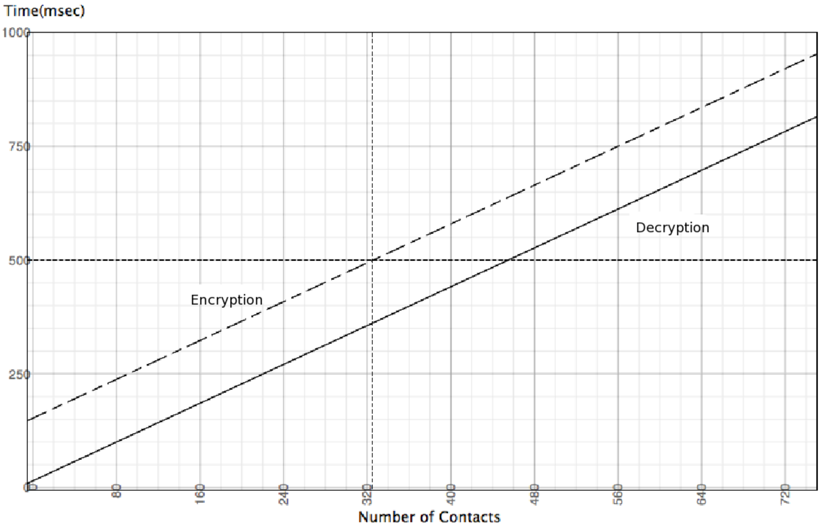


Figure 15: Performance of Scramble operations per contact set

1. Publish operation: this operation is affected by the efficiency of the encryption and signing algorithm.

2. Retrieval operation: this operation is affected by the number of encrypted items per page, and by the efficiency of decryption and verification algorithm.

[20] Tests performed on a 2GHz AMD Athlon(tm) XP 2400+, with 1Gb RAM

Usability

Last, to improve the Scramble! demonstrator we decided to test it among users to see if the user interface and overall usability sufficed. We performed some user tests with local Belgian students, and demonstrations during the Future Internet Conference Week[21]. Scramble! was well received in terms of user experience and functionality. It was also submitted to user interface usability tests performed by usability experts at Cure[22] within the scope of the PrimeLife project. However, a more advanced user experience test targeting a larger audiences is left for future work.

2.6 Privacy-Enhancing Selective Access Control for Forums

2.6.1 Objectives

User-generated content in forums may contain personal data in the sense of personal information, personal ideas, thoughts and personal feelings. In contrast to explicit profiles where, e. g., the date of birth is a specified data item saying “12 June 1979”, the same personal data can be stated in a forum post which is tagged with the date of writing and says “I got two concert tickets at my 30th birthday yesterday!”. In the latter case, it may be not that immediately obvious to the user that she has disclosed her date of birth on the Internet.

The disclosure of personal data in forums and other social software is critical from a privacy perspective, however from a social perspective, the disclosure is necessary since the exchange of information, both personal and non-personal, is the key feature of the application and the primary reason for people to use it. Hence, it is not our objective to prevent disclosure of personal data in forums. We rather want people to be aware of privacy and to enable them to more selectively specify to whom they disclose their data. Access control settings of currently available forums are once specified by the provider and cannot be changed by the individual user. Thus, the user can only decide to disclose information to the groups specified by the providers – in the worst case, this means disclosing to the public – or not to disclose anything at all. Since the first option is not preferable from privacy perspective and the second option is not preferable from social perspective, our objective is to develop a user-centred selective access control system for forums. Forum users should be able to protect their privacy through safeguarding their contextual integrity: data that is disclosed before an intended audience, should not spill over into other contexts and hence possibly have damaging consequences. That is, a user who wants to share her happiness about her birthday present with some other users (e. g. other people going to the same concert) should be able to specify appropriate access control rules to her post in the forum.

Another objective is to sensitise users to privacy issues when using social software in general and forums in particular. Therefore, we aim at raising awareness of this issue by integrating according mechanisms.

[21] http://www.fi-week.eu/

[22] Center for Usability Research and Engineering http://www.cure.at/

2.6.2 Introducing phpBB Forum Software and PRIME Framework

To demonstrate how an existing application can be extended with privacy-enhancing selective access control, we have chosen to build an extension for the popular forum software phpBB[23]. Thereby the main principles of the original system should be preserved. As in other forums, in phpBB, content is always structured in a specific way to make it easily accessible, easy to use, and searchable. In the following, we briefly explain the content structure of phpBB platform, which are illustrated in Figure 16.

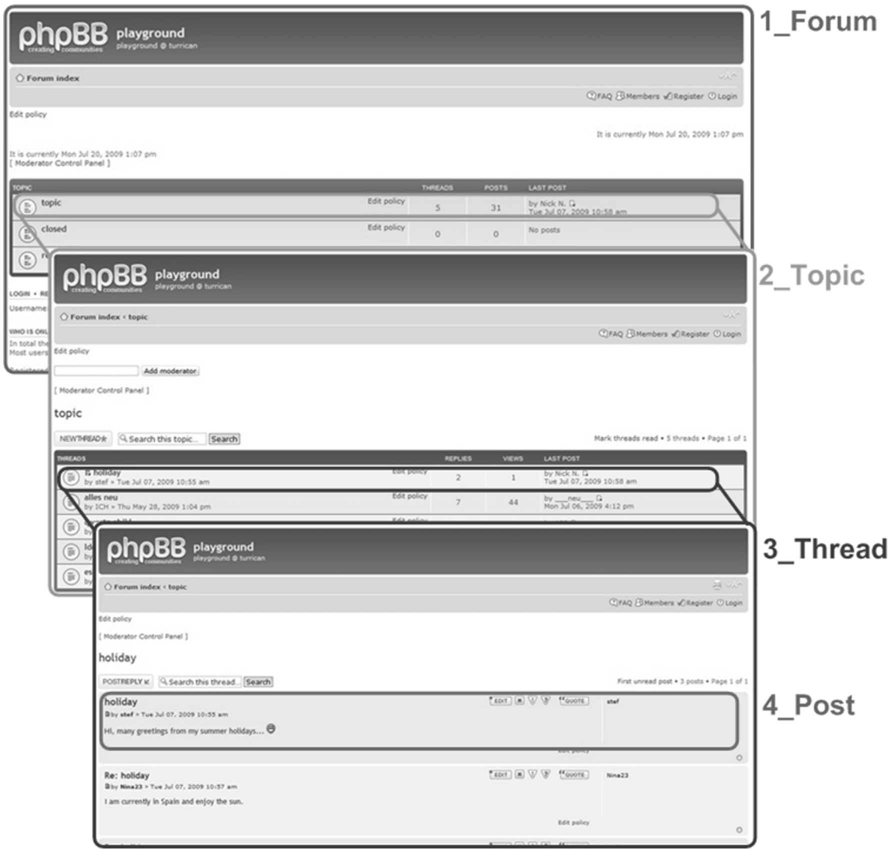


Figure 16: Overview of the hierarchy of a phpBB forum

The top level resource is the forum itself, which is assigned a title and presented to the user when she first enters the platform. An administrator is responsible for managing all general issues of this forum. The forum is subdivided into topics that each address a different subject matter for discussion. For each topic, moderators are responsible for assuring compliance of the content with ethical quality and forum rules. Thus, they have the possibility to change subjects or content of posts, to lock, or even to delete posts. Individual discussions focusing on particular aims are called threads. These are created with the submission of a starting post to which users can reply by submitting replying post.

[23] http://www.phpbb.com

PhpBB software is available with so-called "copyleft license"[24] and is developed and supported by an open source community. This means that the original phpBB source code is available to the public and fairly well documented. With respect to the technical realisation of our selective access control extension, we rely on privacy-enhancing mechanisms that were previously developed in the European project PRIME [PRI]. More precisely, we used *credentials* and *access control policies* from the PRIME framework.

2.6.3 Extending phpBB with Selective Access Control

The most common approaches to access control include the access control *list model*, the *role-based*, and the *group-based* access control approaches. All three require a central instance that defines lists, roles, or groups based on user names, i.e., identifiers of user accounts (cf. [PBP10]). However, social and collaborative interaction in forums does not necessarily require an association of users by their names. Therefore, privacy-enhancing selective access control in forums requires mechanisms that are not based on the knowledge and existence of names or other particular identity information. Furthermore, it is important that users themselves can decide what they deem to be sensitive or intimate information, rather than what is evaluated as such by lawyers, computer specialists or other third parties [Ada99]. This is why we argue that users themselves need to be given control over the audience to whom they disclose data, and hence access control rules need to be set by the user, being the owner of a resource (e. g. a post), instead of by an administrative party. This implies that access control rules should be possible to specify not only for the whole forum or for topics, but also for threads and particular posts. We need to consider that forum platforms typically provide the roles "administrator" for addressing technical issues and "moderator" for content-related moderation of topics. Our approach should allow for keeping both roles. Hence, we can list the following specific requirements for privacy-enhancing selective access control in a forum whereby these are generalisable to further kinds of social software:

- Persons who should or should not be able to access the personal data are not necessarily known by the user.
- These persons also have an interest to protect their privacy.
- Each user should be able to define and modify access rules to her contributions (personal information and expressions of personal thoughts and feelings), i.e., the definition of access control settings should not be restricted to administrative parties only.
- User-controlled and privacy-respecting access control can be applied to different levels of content granularity (e. g. forum, topic, thread, post).
- An administrator of the forum should be able to address technical issues of the platform, but should not necessarily have access to content data.
- Moderators should be able to moderate particular topics.

[24] For more information about the *copyleft license*, we refer to http://www.gnu.org/copyleft/

- The owner of a resource is always able to have access on it.

To address these points in our prototype, we let the user define access control policies together with her contribution whereby an access control policy indicates the properties a reader has to possess and to prove. In order to also protect the privacy of readers, properties are presented in an anonymous way and not linkable when repeatedly used. Therefore, we relied on the concept of *anonymous credentials* proposed by Chaum in 1985 [Cha85] and technically realised in the Identity Mixer (short: Idemix) system [CvH02]. The idea of access control based on anonymous credentials and access control policies is not new in general and was demonstrated in selected use cases for user-service provider scenarios in project PRIME ([ACK$^+$09], [HBPP05]). We built on the results of PRIME, transferred the ideas to social software and demonstrated the practical feasibility of maintaining existing concepts of phpBB platform and integrating privacy-enhancing functionality provided by PRIME framework at the same time.

Using anonymous credentials, everyone can prove the possession of one or more properties (e. g. being older than 18, having more than 10 forum posts with a 5 star rating) without revealing the concrete value (e. g. being exactly 56 years old, having exactly 324 posts rated with 5 stars). In the prototype, credentials are also used to prove the possession of a particular role, which may be required by an access control policy. Thus, in the process of creating a new resource, the originator of that resource receives the corresponding credential (`cred:Owner-`*`Thread-ID`* or `cred:Owner-`*`Post-ID`*) from the forum platform and stores it on the local device. The roles *administrator* and *moderator* are realised with help of the credential-based access control approach as well, i.e., the according credentials (`cred:Admin-Forum` and `cred:Moderator-`*`Topic-ID`*) are issued to the corresponding persons. Together with each new resource, default access control policies are created, which ensure that users who show the administrator credential or moderator credential get the required access granted to fulfill their roles. The owner of a resource possessing the owner credential always has access to that resource and can modify the access control policies to, e. g., allow other users with certain provable properties read access and maybe also write access to the resource.

In general, credentials are offered by particular trustworthy organisations, so-called *credential issuers*. Credential issuers need to be known to the public, so that everybody has a chance to get credentials certifying properties of the user. Moreover, they have to be known (and trusted) so that everybody can verify that received credentials/proofs are valid. More details on the technical implementation of the prototype can be found in [WP110a, WP110b].

2.6.4 Scenario Revisited

Returning to the forum scenario from Section 2.2.2, the following alternative story illustrates how credential-based access control and access control policies in a web forum works. Assume Hannes posts a message to the thread “Online Gambling” in a publicly accessible forum. The access control policy of the thread is derived from the parent topic, which is set to be open for reading and writing exclusively for people who have proven to be registered to an online gambling website. Hannes additionally restricts access to his post to allow only gamblers who have been registered to their site for at least 3 months.

Table 2: Example of an access control policy

(1) Forum:	[(cred:Admin-Forum) OR (everybody*[default]*)] AND
(2) Topic:	[(cred:Moderator-GamesCorner) OR (everybody*[default]*)] AND
(3) Thread:	[(cred:Moderator-GamesCorner) OR (cred:Owner-OnlineGambling) OR (cred:memberOfGamblingSite)] AND
(4) Post:	[(cred:Moderator-GamesCorner) OR (cred:Owner-PostFromHannes) OR (cred:countMonth-memberOfGamblingSite > 3)]

Whenever someone requests access to Hannes' post, the access control policy is evaluated according to the hierarchical order of content elements of the forum (cf. Table 3). In our example, step (1) ensures that authorised users are either an administrator of the forum or – since we have chosen a public forum for the example – any regular user. Step (2) specifies that users are allowed to read the topic "Games Corner" if they are a moderator of this topic or anybody else. The latter applies since the example does not specify any restriction on topic level either. Step (3) ensures that only users who are either moderator of the topic "Games Corner" or who are owner of the thread or who are member of a gambling website get read access to the thread "Online Gambling". Lastly, step (4) determines that only users who are either moderator of the topic "Games Corner", owner of the post, or member of a gambling website for at least 3 months can read the post created by Hannes. Note that read access to Hannes' post is only granted if the whole policy (steps 1 – 4) is evaluated to be "true". Similar to this example for *read access*, further policies can be defined in order to specify *add, edit* or *delete* rights of a resource. All users who add a post to a particular thread have the opportunity to further restrict access to their own contribution.

2.6.5 Privacy-Awareness Information

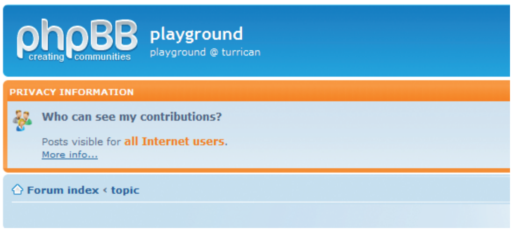
Having described the realisation of the privacy-enhancing selective access control extension so far, in the following we introduce a feature that supports users to be aware of their privacy in the forum. More precisely, we developed a phpBB modification (short: MOD) called "Personal Data" and integrated it into the forum prototype. The idea behind the Personal Data MOD is to provide additional privacy-related information in social software in order to raise users' privacy awareness, help them to better assess their potential audience, and eventually enable them to make informed decisions whether or not to disclose personal data on the Internet. The perception of privacy in social settings depends on the anonymity or identifiability of the users on the one hand, and on the available audience, i. e., who may read and reuse the disclosed personal data, on the other hand. Considering that privacy is only a secondary task for users, presented privacy-awareness information should be easy and quick to understand and not hinder social interactions and communication, which are the primary tasks of social software.

The Personal Data MOD contains two categories of privacy-related information:

Audience Hints about who may access user-generated content, e. g., number and/or properties of potential readers. This partly compensates for missing social and contextual cues in computer-mediated communication [Dör08] and reminds users of the potential mass of "silent" forum readers.

Identifiability Hints about potentially identifying data that is additionally available, e. g., IP address or location information known to providers. This shows that users are not completely anonymous on the Internet and in particular in phpBB forums as there are identifiers available.

In the prototype, the hint about the potential audience is coupled with the setting of the access control policies for read access. If no particular policy is specified for the corresponding forum element and the default policy of the upper-lying content element(s) states "allow everybody", then the Personal Data MOD indicates "all Internet users" as the potential audience for this post (Firgure 17). However, if an access control policy is set which restricts the potential audience, the MOD makes users aware of this fact as illustrated in Figure 18.

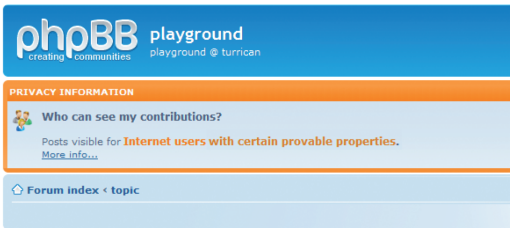


Who can see my contributions?

Posts visible for all Internet users.

More info...

Figure 17: Screenshot prototype: Privacy-awareness information about potential audience if access is not restricted.



Who can see my contributions?

Posts visible for Internet users with certain provable properties.

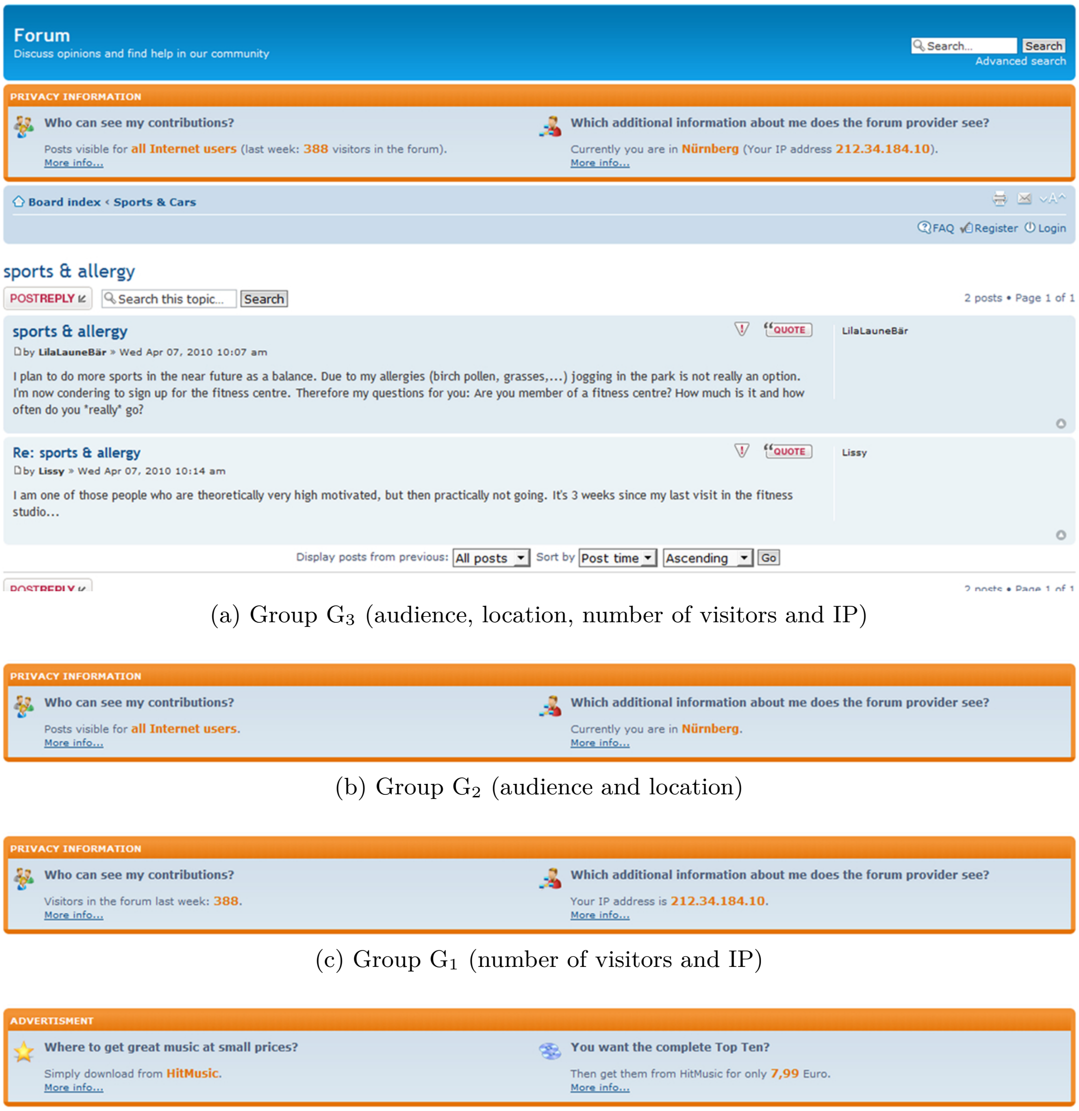
More info...

Figure 18: Screenshot prototype: Privacy-awareness information about potential audience if access is restricted.

2.6.6 User Survey

Besides working on the implementation of the prototype for privacy-enhancing, selective access control, we wanted to evaluate whether our approach meets real forum users' needs. Therefore we conducted an online survey. The survey was available in German and consited of two parts: First, participants saw a realistic full-screen screenshot of the phpBB forum prototype with two posts in a discussion thread about leisure-time physical activity as shown in Figure 19.

We instructed participants to imagine that they are the author of the first of the two contributions. An orange box on top contained either privacy-awareness hints or an advertisement. In the case that privacy-awareness hints were shown, participants saw either textual information about the potential audience and their individual current location, or numerical information about the exact number of visitors of the forum within the last week and their individual IP address, or a combination of both. All participants of the survey were randomly assigned to one of the four groups (G_0, G_1,



PRIVACY INFORMATION

Who can see my contributions?

Posts visible for **all Internet users** (last week: **388** visitors in the forum).
More info...

Which additional information about me does the forum provider see?

Currently you are in **Nürnberg** (Your IP address **212.34.184.10**).
More info...

sports & allergy

sports & allergy

by **LilaLauneBär** » Wed Apr 07, 2010 10:07 am

I plan to do more sports in the near future as a balance. Due to my allergies (birch pollen, grasses,...) jogging in the park is not really an option. I'm now condering to sign up for the fitness centre. Therefore my questions for you: Are you member of a fitness centre? How much is it and how often do you *really* go?

Re: sports & allergy

by **Lissy** » Wed Apr 07, 2010 10:14 am

I am one of those people who are theoretically very high motivated, but then practically not going. It's 3 weeks since my last visit in the fitness studio...

Display posts from previous: All posts ▾ Sort by Post time ▾ Ascending ▾ Go

(a) Group G_3 (audience, location, number of visitors and IP)

PRIVACY INFORMATION

Who can see my contributions?

Posts visible for **all Internet users**.
More info...

Which additional information about me does the forum provider see?

Currently you are in **Nürnberg**.
More info...

(b) Group G_2 (audience and location)

PRIVACY INFORMATION

Who can see my contributions?

Visitors in the forum last week: **388**.
More info...

Which additional information about me does the forum provider see?

Your IP address is **212.34.184.10**.
More info...

(c) Group G_1 (number of visitors and IP)

ADVERTISMENT

Where to get great music at small prices?

Simply download from **HitMusic**.
More info...

You want the complete Top Ten?

Then get them from HitMusic for only **7,99** Euro.
More info...

(d) Group G_0 (advertisment)

Figure 19: User interface for different survey groups (originally shown in German)

G_2, G_3). For the second part of the survey, all participants were shown the same online questionnaire. The questionnaire contained questions about knowledge of technical- and privacy-related terms, use of the Internet in general and of forums in particular and questions related to audiences and access control. We also collected demographic data. A link to participate in the survey was distributed via blogs, mailing-lists and forums on the Internet. Due to this setup, we had a non-random sample as a basis for further analysis. After excluding answers from non-users of forums[25] and from participants who had not seriously answered the questionnaire, 313 valid responses remain. In the following, we report selected relevant results based on those 313 participants. More

[25] Participants who stated in the questionnaire that they have never even read in a forum are considered as non-users.

details about methodology, analysis and further results are provided in [PWG10].

First, to test participants' knowledge and awareness of the potential audience, we asked them who actually has access to "their" forum post that they had seen previously. Second, since we were also interested in participants' ideas and requirements regarding access control, we further asked who they would *designate* to have access. Actually, the forum post that we showed to the participants was accessible for all people with access to the Internet, i. e., all registered and unregistered users, forum providers and Internet providers. The fact that the post from the example was completely public could be learnt from the privacy-awareness display with the textual information (shown for G_2 and G_3) and there was also a visual cue visible for participants of all survey groups indicating that the post can be viewed without being logged in, i. e. it is visible for everybody with Internet access.

Table 3: Expected vs. intended access to forum posts by different given audience groups

	G_0	G_1	G_2	G_3	all
Audience groups	n=78	n=74	n=86	n=75	n=313
All registered users					
expected	96.15 %	97.30 %	95.35 %	97.33 %	96.49 %
intended	89.74 %	100.00 %	96.51 %	96.00 %	95.53 %
Unregistered users					
expected	69.23 %	70.27 %	70.93 %	78.67 %	72.20 %
intended	28.21 %	31.08 %	27.91 %	36.00 %	30.67 %
Forum provider					
expected	98.72 %	95.95 %	95.35 %	94.67 %	96.17 %
intended	66.67 %	75.68 %	75.58 %	70.67 %	72.20 %
Internet provider					
expected	47.44 %	47.30 %	52.33 %	50.67 %	49.52 %
intended	7.69 %	10.81 %	12.79 %	12.00 %	10.86 %

multiple choice questions with given answer categories

A comparison of the percentages of expected access vs intended access of different audience groups, listed in Table 4, reveals that nearly all participants know about and agree with the access to all post for registered members. Also nearly all participants know that the forum provider has access and three-quarters stated that the forum provider should have access. Our results further show that the majority of participants knows that also unregistered visitors can see the post, however only about one-third would *want* unregistered people to view their posts. Hence, there is a considerable difference between the percentage of participants who would let registered users read their posts and those who also would allow unregistered users access to their posts. This finding is interesting for two reasons: First, in most forums on the Internet, anybody can easily become a registered member by providing a fake e-mail address and choosing a password. This means that practically each Internet user could have access with no great effort, anyway and from this point of view there exists no essential difference between registered and unregistered users. Second, the result indicates that participants want to differentiate who can access their forum posts and that their requirements do not match with current access control settings which are defined by providers or administrators. Looking at the figures for the particular survey groups, we found no statistically significant differences between them.

Besides deciding which of the four given audience groups is intended to have access to their forum posts, participants of the survey could specify other groups or share their thoughts about how access control should work in an additional free text field. Indeed, a dozen of the subjects took this opportunity to formulate ideas and said that they would like to authorise particular readers based on special properties or their relationship to them. A selection of comments, which underline real forum users' needs for selective privacy-enhancing access control, is presented below.

> *Selection of comments from participants to the question "Who would you intend to access your forum contributions?" (originally posted in German):*
>
> **C1:** "persons I have chosen"
> **C2:** "authorised by me"
> **C3:** "circle of people that I have defined"
> **C4:** "members with at least 10 posts in the forum"
> **C5:** "friends"
> **C6:** "guests who I have invited"

These requirements can be addressed with the selective access control extension for phpBB forums. The extension enables users to define which properties someone has to possess for gaining access to users' contributions. In order to allow for proving properties related to private life, it is even conceivable that users themselves issue credentials which provide evidence of an existing relationship, e. g. *being friends in community X*. Thereby it is also possible to realise relationship-based access control with the selective access control extension. As argued previously, access control based only on relationships is not suitable for forums in general since this requires that the author of a contribution and the users she wants to give access have to know each other before. This assumption does not hold for web forums, where people with similar interests can meet and discuss without knowing each other in person.

Chapter 3

Trustworthiness of Online Content

3.1 Introduction

Some decades ago, relatively few institutional sources such as Encyclopedias provided content that people based important judgments on. In the 1990s, the Internet started to replace traditional media as an invaluable source of information. Ten years ago, web content was provided by a limited number of institutions or individuals, however, today's Web 2.0 technologies have enabled nearly every web user to act not only as consumer, but also as producer of content. Indeed, many services available on the Web are fundamentally based on user contributions. A prominent example are wikis such as Wikipedia which are entirely based on content contributed by multiple users and modifiable at any time by any of them.

Individuals and organizations increasingly depend on this type of distributed information with its severe trust limitations. In the Web 1.0, it was already difficult to decide to which extent online sources could be trusted. With Web 2.0, the question of trust in online content becomes central: Users cannot be sure whether an information is correct, whether the information will be accessible in the future, whether it is legal to use it, and who would assume liability in case the information is incorrect. Users of the Web are not protected against lies and misinformation - think of the recent cases of intentionally false articles in Wikipedia (e.g., BBCNews[1]), or stock price manipulations through misleading newsgroup postings (e.g., CNet[2]).

In fact, with the highly dynamic information flow enabled by the Web, information often takes a life of its own as it can, for example, be published, edited, copied, aggregated, or syndicated; it eventually becomes detached from the context in which it was created and evolves separately. Users do not have cues to determine whether they

[1] UK politicians' Wikipedia worries, published Friday, 6 March 2009, accessed 16 Sept. 2010, http://news.bbc.co.uk/2/hi/uk_news/politics/7921985.stm

[2] Bounty offered for stock tipster, C'Net news, September 5, 1997 12:15 PM PDT, accessed 16 Sept., 2010, http://news.cnet.com/2100-1023-202979.html

can trust the information or not. Personal ad-hoc strategies to deal with this, such as trusting only certain websites, are no longer appropriate when the information is dynamically assembled from multiple sources or is passed around and republished, or when the website itself does not perform editorial tasks but entirely relies on its users.

In general, the term *'Trust'* is a difficult quality to define precisely because there are often implicit qualifiers which are determined by the context in which the term is used and also it is used to mean many different things (even within a single context). Definition is further hindered by the gap between *'trusted'* and *'trustworthy'*. When speaking about *'trust in content'* we adopt a limited scope and take this to be *'the belief that the information is true, accurate or at least as good as possible and the reliance upon this belief'*. When speaking about *'trustworthiness of content'* we mean that *'the content satisfies a set of conditions (with implicit qualifiers) defined by another party to justify her well-founded trust in the content'*.

Trust in content is most often derived from trust in a person, entity, or process. As such, there needs to be a binding between content and the person, entity, or process from which trust may be derived. There are two standard approaches to address this binding. The first, more commonly used, consists of trusting the deliverer of the content. These include online news sites, corporate Web sites, and main entries in a blog. The second approach is to include meta-information along with the content that the user may use the assess properties of the content. Such meta-information includes for instance (information about) the originator, author, reviewers, the process of how the content was produced, or the links to external authoritative sources.

The point of *'trust in content'* is enabling consumers to assess (correctly) the *trustworthiness of content*. Such enabling involves a combination of technical mechanisms, psychological insights, and user education.

Work on technical mechanisms and findings of psychological insights derived by experiments are described within the remainder of this chapter. We first investigate the scenarios of wikis/blogs and derive requirements for technical mechanisms in Section 3.2. As users have both the wish that content can be trusted and the wish of protecting their own privacy a balance needs to be made between both requirements. Considering this, we developed technical mechanisms that do not attempt to make fully automatic trust decisions on behalf of users. Instead, users are presented relevant (primary and secondary) information in such a way that they can conveniently and efficiently interpret that information as part of the mental task of finally deciding about the trust level of the given content. This act of interpretation can then include whatever additional subjective considerations users wish to apply. Which consideration users apply was studied in experiments we present in Section 3.3. What these experiments basically have shown was that users need education to fully use the possibilities the Internet offers them to establish trust. Finally we present some technical solutions in Section 3.4 that try to aid users in forming their human trust decisions; they do not replace or incapacitate them and should all come along with user education.

3.2 Scenarios and Requirements

This section sketches real world scenarios of user-contributed content based on wikis and blogs. These scenarios demonstrate the kind of use of online content in different situations as well as the role of trust into the content in these settings. This provides the starting point for exploring a set of mechanisms that allows for the realization of these and similar scenarios. From the mechanisms, a number of more detailed requirements can be derived that were used to build the prototypes documented later on in Section 3.4.

3.2.1 Scenarios

There are numerous possible scenarios available on the web that have a need for trusted content. We selected wikis and blogs as application domains since they are very popular and provide comprehensive functionality to work with user-generated content. This enables ordinary users to intuitively manage the tasks of creating, retrieving, and assessing content. Thus, both of them are ideal candidates for conducting according experiments.

Blog

A blog is a sequence of generally short articles, often called entries, published on the Web. The entries are produced by an individual or collection of individuals who are the blogs author/s and are often connected by a theme. The entries may consist of text or multimedia, as in a video blog (vlog) or a photographic blog (photoblog). We interpret the term blogs in a relatively broad sense, i.e., not just encompassing individuals online journals, but all content that is 'pushed' over RSS or Atom protocols, and other similarly structured content (e.g., from electronic mailing lists) that is easily transformed into the common format. The popularity of blogs as a medium derives from the low cost of production and distribution of content. A direct consequence of this is a large base of content producers and of topics addressed. The issue of whether online information can be considered trustworthy is especially urgent when new information arrives that has to be acted on quickly such as blog articles that convey important news. Often these articles are published by an initially unfamiliar source of origination. Examples are:

In-company weblog: Employees of a multinational company (or members of another organization) consume news from similar sources in the form of blogs. Not each individual participant may have the capacity to judge each piece of information in its self-contained form (for a start, it may be phrased in a foreign language), yet the entire 'crowd' of members can form an enhanced overall view for 'inside' members on augmented 'outside' information. This scenario was investigated with a demonstrator outlined in Section 3.4.1.

(Medical) selfhelp: Private individuals who consume health-related information (e.g., consider treatment options adjacent to interviews with their physicians), and have obvious warranted interest that this information be trustworthy (e.g., 'Is it advertisement? Is it a rumor? What does my insurance company say about it? What is it?'). This scenario points to the importance of including privacy-friendly trust establishing technology. Experience shows that individuals somewhat differ in their judgments as to the most desirable and practical levels of privacy [Bri98] based on cultural background, politics, age, and other factors; yet privacy is generally held an indisputable right and value when it comes to information that concerns personal health. We use this scenario to investigate which meta-information consumers concentrate on, when making trust decisions on health-related information as outlined in Section 3.3.1.

Wiki

A wiki is a collection of Web pages, which can be edited online by its users. The objective of a wiki is to support users in collaborative creating and editing of common shared contents. It is possible to link content and to comment on it. Wikis also provide history functionality so that it is easily possible to reset pages to former versions. While some wikis are accessible and editable without authentication, others have fine-grained access control. A problem of information published in wiki systems is that information can easily be manipulated or tampered with. An advantage of a wiki system is that information can easily be corrected. The weakness of the system is therefore also its strength, at least if the user base is sufficiently large, committed and knowledgeable. To prevent misuse or vandalism, most wikis try to adopt the strategy of making damage easy to undo rather than attempting to prevent it in the first place. A widely known example for this is the history function with which one can restore pages to older versions. One of the major difficulties in wikis is that it is hard to establish whether information is reliable. The reliability of information depends on the author(s) of the content. Wikis may therefore adopt different mechanisms to control who can create and edit information. One mechanisms is that of captchas in conjunction with a text edit field. A captcha is a type of challenge-response test used to ensure that the response is not generated by a computer. Other mechanisms introduce a waiting period before an editor can contribute to a wiki which aims at preventing spur of the moment modifications. The English Wikipedia, for instance, requires new users to wait at least four days before they can contribute. This prevents, or at least delays, rogue automated programs to make edits. Another example is the Portuguese Wikipedia where a user has to make a certain number of edits to prove her trustworthiness and usefulness as an editor. The German version of Wikipedia is currently testing an extension called Flagged Revisions which lets trustworthy authors assign sighted or certified tags. The conditions for an author to be able to assign sighted tags are restrictive in days of having an active account as well as number of edits. The certified tag is work in progress.

We investigate this scenario both with a prototype in Section 3.4.3 and a related experiment described in Section 3.3.2.

3.2.2 High-Level Mechanisms

By providing consumers with a technological means for not only viewing the primary information online but in the context of related assessments by others whom they are acquainted with, and who in turn may be better acquainted with the primary information, we can facilitate more educated trust decisions that are of benefit to consumers. Trust is ultimately a personal decision. Different individuals may make different choices even when presented with the same ‘objective’ evidence, and not everybody is able or even willing to express what exact considerations go into their respective trust decisions.

Core mechanisms

On a functional level, technical mechanisms assist users in such a way that they ensure the relation of meta-data to particular content, the authorship (in an absolute or pseudonymous sense), and that it has not been tampered with. We distinguish the following mechanisms on a functional level:

Evaluating trustworthiness: This refers to the process of condensing all available meta-data (such as ratings) that belongs to a piece of information. It forms part of a process that ultimately leads to a binary trust decision on the information whose trustworthiness is under consideration. As an intermediary step, a numeric score for a piece of content may be calculated automatically, which users may then base their final judgement on.

User reputations and certifications: The assessment of trusted content depends on who provided a piece of information (or who provided secondary information about it). Users can collect so-called certifications and ratings that are aggregated to a reputation. User reputations serve to characterize users in ways that are meaningful to other users when it comes to judging them as sources (e.g., highly reliable source on a scale from 1 to 5).

Binding metadata to data: The trust model assumes that when a user does not know whether to trust some piece of information, she can triangulate by taking other, secondary information (meta-data, such as ratings) from other users into account. This entails mechanisms for strong bindings of information to its pursuant meta-data.

Supportive means

Beneath functional mechanisms users need supportive means to deal with the meta-data they got and to provide this meta-data to others:

Privacy-friendly incentive system: Not always are users self-motivated to contribute and other incentives are needed to get them to contribute. An incentive system enables users collect and offer some digital points (which can be translated into money, mp3 files, etc) in exchange for contributions. Of course, such a system must offer at least the degree of privacy the collaborative platform itself offers.

Trust policies: These are artifacts that allow users to declare conditions on meta-data (such as ratings or scores) in order for the information to be regarded trustworthy, or before information can be transformed during its life-cycle.

Anonymous networks: Traditional electronic communication channels reveal by construction lots of information about the communication partners to the partners themselves and also to third parties providing the communication infrastructure. To be able to control what information the communication partners reveal to each other, the communication layer must be "anonymous." There are a number of known solutions that provide anonymous and secure commnication.

3.2.3 Requirements of Mechanisms

In this section, we make a first iteration of detailing the core mechanisms described above. From the supportive mechanisms we chose privacy friendly incentive system to do this iteration.

A pre-requisite for all mechanisms are the following requirements:

- Open API: The system should offer external interfaces that respect relevant open standards for web and service interfaces such that it can be coupled to existing applications in a straightforward manner. In the case of web applications, this can have obvious advantages in terms of potential size of user community, remote access, etc.
- Efficiency: The system should employ an efficient representation of its (cryptographic) artifacts, both in terms of their in-memory representation and resulting requirements on surrounding protocols.
- Scope: The system should be applicable to a wide range of target applications, e.g., by using a decomposition into (a smaller group of) components that are specific to a (set of) application(s) and (a larger group of) general components that can serve any target application.

User reputation and certification

A user reputation and certification system comprises of the following mechanisms:

A user-rating mechanism: While our primary focus is content, users also can be ranked (providing them a reputation). This mechanism allows a party or process to specify the atomic rating of an individual or organisation (who/which produced the content).

A rating aggregation algorithm: A rating algorithm aggregates individual ratings to one single reputation. It may allow weighing a set of individual object or entity ratings which would require weight factors to be specifiable/specified.

Certification: When entities rate content, the relying parties should be able to trust that the ratings are actually provided by legitimate raters. Certification of the raters can warrant for this property. Certificates are basically digital signatures issued by a third party that is trusted by the user and that verifies that a public key is owned by a particular party.

Web of Trust: This is a decentralized concept based on recommendation that is used to establish the authenticity of the binding between a public key and a user, for example, a PGP trust model. A network of trust can be generated using such a model. This can be contrasted with the centralized or hierarchical relationship between certification authorities that exists in X.509[3].

[3] For more information, refer to http://www.itu.int/rec/T-REC-X.509/en.

A mechanism to propagate ratings: Ratings are propagated over some kind of rating network. A mechanism which models this network and the message exchanged is needed for rating systems.

A mechanism to store reputation: There are different ways to store reputations, e.g., it may be stored decentrally on user side or centrally at a reputation server.

These mechanisms have to meet the following requirements:

Authentication of parties: Users want to both demonstrate that they can be trusted and also ensure that the parties they deal with are trustworthy.

Completeness of reputation: Users want the aggregated reputation to consider all ratings given. During the storage and propagation of reputation it should not be possible for the entities involved to omit certain ratings.

Pseudonymity of authors and raters: Users want to rate and provide web content under a pseudonym to not necessarily allow others to link this rating to their real name. In the real world there are also authors who write under a pseudonym and many services in the Internet also allow the use of pseudonyms instead of real names following EC Directive 95/46 [95/46/EC].

Anonymity of users: Users want to evaluate reputation anonymously to prevent others from building personal behavior profiles of their possible interests.

Persistence of reputation: The same reputation should be available for all pseudonyms a user uses in a context.

Self-determination of shown reputation: If there exist only few authors with the same reputation these authors are easily linkable despite of using different pseudonyms because of the same reputation value. Thus, authors should get the possibility to determine how much of their positive reputation they show. Negative reputation must not be omitted.

Transparency: The reputation algorithm must be able to show how an aggregated rating was derived on the basis of individual ratings. The system has to be designed in a way, that the user may check the integrity of single ratings as well as the integrity of the reputation.

Binding metadata to data

Secure metadata support comprises the following mechanisms:

A mechanism for combining a piece of data and its metadata in a secure manner: This mechanism ensures that content and the meta data remain associated, that is that it is impossible to tamper with the content without this being reflected by the meta-data. This can, for instance be achieved by forming signature of the whole by the originator of the data.

A mechanism for checking that metadata belongs to its data: This mechanism allows the relying party to check whether the metadata actually concerns the data to which it is purportedly associated. This could be accomplished by offering the relying party a mechanism for checking the signature of the combined bundle.

A mechanism for reliably referring to single instances of a piece of data:

These mechanisms have to meet the following requirements:

Integrity: The system must ensure that the combined bundle of data and its metadata is safe from unauthorized modification.

Non-repudiation: The system must ensure that the effective combination of data and its metadata cannot be denied by a user who created a signature on the bundle. This requirement may conflict with the requirement of authors being able to contribute and rate pseudonymously in the system. The harmonisation of these two requirements requires special attention.

Normalization: The system should be able to normalize both data and meta-data to account for the fact that (semantically) equivalent forms may be represented by different byte strings, yet should lead to same signature values.

Transparency: The mechanism for reliably referring to single instances of a piece of data should respect existing conventions for data references in general. (This can, e.g., be achieved by forming URLs under a new schema.)

Evaluating trustworthiness (or any other property of content)

A trust evaluation system for content comprises the following mechanisms:

A mechanism to request content to be evaluated: This mechanism allows a user to specify that certain content needs to be evaluated in terms of trustworthiness, integrity, validity, relevance, etc. The requester may associate a reward or incentive to the fulfillment of the request. The incentives or rewards may be specified in terms of privacy-friendly incentive points (see supportive measures).

A rating mechanism: This mechanism allows a party or process to specify the atomic rating of particular content (i.e., the content-rating). The rating may be based on the entity-reputation of an individual or organisation who/which produced the content, on certain qualities of the content (content-rating) as assessed by the rater or the rating process (in the case of e.g., text analysis).

An aggregation algorithm: A content rating aggregation algorithm aggregates individual ratings to one single content quality rating. It may allow weighting of single ratings based on a set of individual content ratings which would require weight factors to be specifiable/specified.

A mechanism to propagate ratings: Ratings are propagated over some kind of rating network. A mechanism which models this network and the message exchanged is needed for rating systems.

A mechanism to store ratings: Most likely the content ratings are stored on the content server.

Similarly to the user reputation and certification system these mechanisms have to meet the following requirements:

Availability of reputation and ratings: As a functional requirement, each user of the rating system should be able to access reputations and ratings to estimate the quality of web content.

Integrity of web content and ratings: Users want web content, ratings and reputation to be preserved from manipulations, both in propagation and in storage.

Accountability of authors and raters: Users want a content's authors and raters to be accountable for the web content they provided respectively rated. This requirement may conflict with the requirement of authors being able to contribute and rate pseudonymously in the system.

Completeness of reputation: (same as in user reputation)

Pseudonymity of raters: (same as in user reputation)

Unlinkability of ratings and web content: Users want to rate and provide different web content without being linkable. Otherwise behavior profiles of pseudonyms (e.g., time and frequency of web site visits, valuation of and interest in specific items) could be built. If the pseudonym can be linked to a real name the profile can be related to this real name as well.

Anonymity of users: (same as in user reputation)

Confidentiality of ratings: Although a reputation system's functional requirement is to collect and provide information about a reputation object, raters might prefer to provide only a subset of their ratings to a specific group of other users while keeping it confidential to all others.

Liveliness: The system may allow existing content ratings to be replaced by novel ratings. This may even be required on the basis of new information, for instance when a rater turns out to have provided unwarranted ratings.

Privacy-friendly incentive system

A suitable privacy-friendly incentive system comprises the following mechanisms:

Obtaining privacy-friendly incentive points for circulation: This mechanism allows users to obtain a collection of privacy-friendly incentive points from a reserve. Points in this collection will effectively enter circulation, and the reserve will enforce an overall policy on the flow of incentive points (e.g., maximum issued number linked to monetary equivalents in users' accounts).

Exchanging privacy-friendly incentive points: This allows a party to offer incentive points for certain online transaction and to transfer those points to another party once a transaction has occurred.

Removing privacy-friendly incentive points from circulation: This allows parties to return privacy-incentive points to a reserve. Such points will be withdrawn from circulation, and the submitting user will typically receive some suitable form of other compensation (e.g., monetary deposit to her account).

These mechanisms have to meet the following requirements:

Pseudonymity: Users must be able to offer and receive privacy-friendly incentives under pseudonyms and without the need to reveal their real identities.

Double-spending: The system must be able to detect when users try to cheat by spending the same privacy- friendly incentive points on multiple parallel occasions (i.e., they overspend). Double spending must lead to certain disciplining behavior, such as revealing the users identity to warn against future misuse.

Accountability: It must be possible to hold parties accountable the actions taken within the scopes of defined mechanisms. For instance, this must be true with regard to the exchange of pending privacy-friendly incentive points, or with regard to disciplining users because of their alleged double-spending.

Unlinkability: The system must ensure that uses of different privacy-friendly incentive points remain unlinkable, as long as they spending them responsibly (i.e., do not overspend).

Off-line: The system SHOULD support off-line use of privacy-friendly incentive points, i.e., two users can exchange such points without a central party (typically the reserve who issued points in the first place) having to become involved. Especially in an off-line scenario it has to be ensured, that double-spending is not possible.

Distribution of concerns: The incentive system should allow parties to store their digital artifacts (e.g., privacy- friendly incentive points) locally, and should not introduce unnecessary assumptions for central storage or other processing at a single location. In case of local storage of the digital artifacts, loss of these artifacts is a concern. Should the system be capable of re- issuing lost credits?

3.3 Experiments

In the previous section requirements and mechanisms were sketched that may be used to help internet users assess the trustworthiness of online content. Before we describe in Section 3.4 how these mechanisms can be implemented technically we evaluate additional requirements from practical user experiments.

The first experiment we describe in Section 3.3.1 relates to ‘Binding metadata to data’ (Section 3.2.3). We want to know which metadata is useful to function as trust markers. Although, as we mentioned earlier, trust ultimately is a personal decision, there are of course patterns and some data are more relevant trust makers than other.

The second experiment we describe in Section 3.3.2 relates to 'User reputation and certification' (Section 3.2.3). Our goal was to find out how private users consider their reputation and other attributes. Based on this we can suggest how to make a trade-off between the metadata other users want to know about content and the trust information others are willing to reveal.

3.3.1 Binding Metadata to Data

The first experiment aimed at better understanding the criteria employed by internet users in order to determine which information to trust and which not to trust.

In order to learn more about internet users' mental trust models and what people consider to be relevant cues regarding content quality and trustworthiness, and how content evaluators handle rating content, we have conducted a few experiments. Questions guiding this research were:

- What are relevant properties to be rated? What are the most salient features of content to call it trustworthy (e.g., validity, accuracy, completeness)? Should the quality be associated to the object-quality score (like author reputation is confined to the domain at hand), or will this be unmanageable by end-users (raters and readers)?
- What are relevant author properties to be rated?
- What binding is required between content and author or rater? Math proofs provided by math professors are likely valued higher than those provided by math students, but this does not say anything about the professor's reputation regarding fast cars.

Related research

Research on credibility assessment of information found online generally demonstrates that factors pertaining to characteristics of the content, i.e., usefulness and accuracy of the information, factors pertaining to authority, i.e., trustworthiness of the source, as well as factors pertaining to the presentation of the information play key roles in people's credibility assessments [Met07, Rie02, EK02, FSD$^+$03, FC01].

However, there appears to be a discrepancy between indicators people believe they use in order to appraise information they find online and indicators they actually use when assessing the credibility of information [EK02]. Internet users typically indicate that their judgements of website credibility is affected by the identity of the source and scientific references. But, results from observational studies demonstrated that people rarely pay attention to these factors and generally spent little time evaluating the quality of information [EK02, FM00, FSD$^+$03]. Instead, it seems that the presentation of information and the design of websites are the most important determinants of internet users' credibility assessments.

On the one hand this finding might be somewhat distressing because people might be easily misled by appealing webdesigns and consequently trust information that may be of little value and low quality. On the other hand, and from a privacy protection point of view, the finding that information about the source has little impact on internet users' credibility assessments implies that information about the identity of the author does not need to be disclosed on websites. If internet users pay little attention to features of the source of the information they read on a website, the absence of this indicator will not interfere with their credibility assessments. Consequently, information lacking author specifications will not be regarded as less credible.

This assumption is counterintuitive, especially when keeping in mind that people believe that information about the source is the primary factor influencing their credibility assessments. To explore this assumption in more detail, we have conducted an experiment in which we tested which indicators determine whether or not people find information trustworthy and what role author specifications have in this process.

Experiment and findings

The experiment and questionnaire were designed to explore which indicators people view as credible when searching information on a wiki. The test subjects were presented with a mock-up of a medical wiki that had to be used to answer questions about an unfamiliar topic (5-HT or Serotonine). The test subjects could enter search terms in the wiki in order to find information to answer the questions. After entering a search term into the wiki, participants received a list of search results in a random sequence. The list consisted of six search results, each providing a few random words about 5-HT along with information about the author of the text (trust indicators). Each of the results bore one of the following trust markers: (1) the name of the author, (2) the title and name of the author, (3) the occupational title of the author, (4) the title and reputation of the author, (5) a certification of the author, and (6) a reputation measure for the website.

Participants could then choose one of the hits and received a text that contained information they could use to answer all questions about 5-HT; the text was slightly different for every hit but included the exact same information. Each text was associated to one indicator, i.e., for each subject the text presented for a given indicator (such as name of the author) was always the same.

After having received the search list generated by the wiki, subjects could select a hit from this list and read it and then return to the search results list to choose other hits. In addition, they were free to enter as many new search queries as they wanted. Each time participants returned to the list with search results or entered a new search query, they received four questions concerning the expertise and trustworthiness of the source and three questions referring to the quality of the information using 10-point Likert scales with items: competence, knowledgeability, objectivity, truthfulness, accuracy, usefulness, and comprehensiveness of the author).

The procedure therefore was: select hit, read text, answer the 7 information credibility questions and return to the search results list to repeat the procedure, or answer the 3 questions concerning 5-HT.

After submitting their answers to the 5-HT questions, subjects received a questionnaire about search strategies. The purpose of this questionnaire was twofold. First, one question about reasons for selecting one or more additional hits during the task was integrated into the questionnaire as an additional credibility measure. Second, the questionnaire was designed to generally measure people's strategies for searching information on the internet.

The test subjects in the experiment (256 students at Tilburg University, TU Dresden, National University Ireland Galway, and ETH Zurich, resulting in 172 useable response sets) appear to favour the first search result in the search results list, irrespective of the source of the information. The findings of the experiment demonstrate that internet users' credibility assessments are mainly influenced by the position of information in a search list. The position of a hit in a search list was the most important indicator followed by information about the occupational title of the source. Personal information about the identity of the author was not a particularly relevant indicator for trustworthiness, at least not when compared to position in the search list and occupational title of the author. Personal information about the author, such as her name, became a more important indicator as people selected more than three hits from the search list. Information about the occupation or reputation of the author are more relevant than her name. In addition to these indicators, a reputation measure of a website was found to influence people's credibility assessments, whereas a certification such as the one used in the present study (author is a member of the American Medical Association), does not seem to be a valuable indicator for credibility.

When looking more closely at what the subjects say about the quality of the information it appears that the test subjects believe that information that is provided along with the occupational title of the author has a higher quality than information that is provided along with a certification of the author regardless of the actual quality of the information. It also appears that position of a hit in a search list generated by a search engine is the most important indicator for its trustworthiness for people's first search, whereas indicators providing information about the source become more important than the position for the subjects' subsequent searches. The main reason for participants to visit more than one search result was to look for confirmation of the information already read on the first entry.

While these findings demonstrate that people have a strong tendency to rely on the position of a hit in a search list, they indicated that, in general, professional writing, message relevance and the absence of typos were the most important indicators for trustworthiness of information they found online. In line with the findings, participants indicated that, in general, information about the source was of relatively little importance in terms of credibility. Actually, presence of contact information, author identification and author qualifications and credentials were rated as least important indicators for reliability on information found online. Interestingly, only 17.9% of the participants indicated that the position of a search result in the list was very important. When comparing the actual behaviour of the subjects with their beliefs, it becomes clear that people believe the position of a hit in a search list is less important for their decision to select a hit than it actually is. This discrepancy between indicators people believe they use in order to appraise online information and indicators they actually use when assessing the credibility of information is in line with previous research findings. However, in contrast to previous findings, participants did not indicate that they found information about the author very important, while this information did affect their actual behaviour, albeit to a lesser extent than the position of a hit.

Taken together, the findings demonstrate that our test subjects believe they base their decisions whether to choose a hit and rely on the information they find on the internet on characteristics of the content (as reported in their answers), while actually, convenience, i.e., the position of a hit in the search engine output, mainly determines their behaviour (as witnessed by the fact that they hardly ever consulted a second search result in the experiment).

3.3.2 User Reputation and Certification

According to [Ada99], it is more important that what is deemed sensitive or personal data is based on the perception of the individual rather than if the data can be evaluated by third parties (e.g., lawyers, computer specialists). Considering that individuals often claim to have a great need for privacy but behave differently (cf. *privacy paradox* [Pöt09]), we decided to conduct a study with an experimental part to learn how users actually treat their own reputation value compared to other personal data items. In the following, we briefly outline the set up of the study and report key results. This experiment is also published as an outcome of PrimeLife in [KPS11].

Study design

The web-based study consisted of an experiment and a questionnaire. Invitations to participate in the study were posted in several forums and blogs on the Internet and we also distributed flyers in the university library. All participants who completed the study were offered the chance to win vouchers for an online shop. For the experiment, all participants were asked to rate the same articles from a wiki about books and literature according to three given categories. Before participants actually accessed the wiki articles, they did a short literature quiz. By answering four multiple choice questions about famous writers and books, they received between zero and four points. These points are considered as a subject's *reputation*. Subjects were further asked to indicate name, age and place of residence. When rating the wiki articles subsequently, each participant decides whether her

- name,
- age,
- place of residence and/or
- reputation

should be published together with her rating of a wiki article.

Half of the participants were randomly assigned to the experimental group. The experimental group were shown privacy-awareness information, i.e., information about who can see what data about the user, together with each wiki article. The other half of the subjects belonged to the control group and did not receive privacy-awareness information.

After finishing this first part of the study, all participants filled in the questionnaire. In this questionnaire, we asked about the perceived level of privacy in the wiki, about experience with wikis, ratings systems and the Internet in general. We used questions from the applied privacy concerns and protection scale [Tad10] to investigate general caution, privacy concerns and applied protection measures. Finally, we asked about demographic data and whether subjects had given their real name, place of residence and age at the beginning.

We calculated the *Perceived Privacy Index* (PPX)[4] from participants' answers to the questions about how public, private, anonymous and identifiable they felt in the wiki. Each item was measured on a 0 to 100 % slider scale. The higher the PPX value, the more private a subject felt.

Results

After excluding complete data sets from a few subjects who admitted not to having seriously participated in the study, 186 valid responses remain and were used for further analysis.

30 % of the subjects agreed to publish their real name together with the rating of a wiki article. The disclosure of their real age was okay for 57 %, real place of residence for 55 % and 63 % agreed to have their reputation value published. This means, for each data item there was a considerable share of subjects who wished to keep this information private. If participants indicated later in the questionnaire that they did not provide true information in one of the first three categories, we treated this data item as not disclosed. Since the reputation value was calculated from answers in the literature quiz, cheating w.r.t. reputation was impossible.

Further, we used a linear regression model to calculate how the disclosure of these four data items and a few other factors influenced user's perceived privacy in the wiki. The results are listed in Table 5 and reveal that there are only two factors that significantly decreased the perceived privacy: the fact that a user has published her name and the fact that a user has published her reputation value. While it is not surprising that a user feels less private after disclosing her real name, we found that also disclosing their reputation value had a similar effect on perceived privacy. According to the results, the reputation value is deemed an even more sensitive piece of data than age or place of residence. Application-independent measures, i.e., privacy concerns, general caution and technical protection measures, did not play a significant role for perceived privacy in the wiki.

Altogether, the results of our study underline that a user's reputation value has to be treated as a personal data item. That means that in a reputation system, users should have the possibility to keep their reputation private, or to disclose only an approximated value.

3.4 Demonstrators

In the following we give brief summaries about the demonstrators we built corresponding to the mechanisms described in Section 3.2.2. The demonstrators can be used for either the wiki or blog scenario outlined in Section 3.2.1.

[4] The questionnaire contained the question "Please indicate to which extent the following adjectives describe your feelings while using the wiki: 0 % (not at all) – [*adjective*]– 100 % (very much)?" (originally asked in German). The PPX is composed of the adjectives "public" (scale inverted), "private", "anonymous","identifiable" (scale inverted).

Table 4: Regression model, *n=186*.

Perceived Privacy Index *PPX* (dependent var.)	Est.	Std.er	p
Intercept	288.93	33.47	
Application-specific predictors			
Privacy-awareness information available	4.57	12.01	0.704
Name published	−46.66	14.49	0.002**
Age published	−13.54	16.77	0.420
Place of residence published	−21.65	16.06	0.179
Reputation value published	−39.99	14.04	0.005**
General predictors			
Privacy concerns	−1.35	1.10	0.223
General caution	0.22	1.79	0.902
Technical protection	−0.47	1.47	0.750

sign. levels: $^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

3.4.1 Trustworthy Blogging

Information found on blogs is often commented upon and referred to by other users in their own blogs or other forums. The main idea of the demonstrator is to collect and present such comments to users when they read a blog article. That is, the readers does not only see the blog article itself but their browser also presents them the comments made elsewhere. Thus, by reading also this secondary information, the reader can base her assessment of the trustworthiness of the blog article on. This is of course requires that the Internet is searched for such comments and (reputation) information about the users who provided theses comments is available.

The demonstrator implements this idea on the IBM corporate intranet. As there exists a central directory that provides information about each employee, the demonstrator focuses on finding and indexing comments and then making available to the users all these comments as well as the information about the originators from the central directory. The demonstrator further offers users the possibility provide their own comments on articles they had read.

Demonstrator architecture and mechanisms

The demonstrator consists of a central server collecting and indexing comments and of components that display information to the users. The latter include a firefox-plugin that apart from displaying a web site, e.g., a blog entry, also displays the related meta-information such as comments and identity information about the commenting users. It also offers reader means to provide their own comments on a read blog entry to the demonstrator server.

The central server provides two main functionalities: It is first a service which readers can query for meta-information w.r.t. a piece of information, i.e., a blog article they are reading. The meta-information include comments on and references to this piece of information the demonstrator has found by crawling the net as well as comments and annotations that readers submit to the central server once they have read an article. The second functionality is to collect these meta-information and to maintain an index.

Most of the mechanisms needed for the implementations are rather straightforward and we do not describe them here. The main technical challenge was to find a mechanism to bind information (e.g., an article) to its metadata (comments but also all other information about the article such as its author or source). Thereby we can in general not assume that pieces of information (text) have a unique identifier such as a URL. To solve this, we have introduced the concept of a bound URI (BURI). A BURI is computed from a duple consisting of information and its meta-data. The computation of a BURI involves *normalizing* the information to give it a unique representation, *versioning* of it, and then *binding* the information and the meta-information together (e.g., by a digital signature by the originator of the information and meta-information).

Learnings

User generally find the presentation of the collected meta-information very helpful to assess blog articles. However, the motivation to offer comments themselves seems to be rather low. To address this, we have developed a privacy friendly incentive system that is described in the next section.

We finally note that a central server that provides users with meta-information can potentially track what articles a given user reads. This could be alleviated by having the users to request this service via an anonymous network. An alternative method is to use mechanisms that hide the query from the server. Also, if the providers of a blog would mirror the related meta-information, this problem would not occur to start with.

3.4.2 Encouraging Comments with Incentives

User-generated content often varies in quality and accuracy. Its trustworthiness as well as its quality can be significantly improved by (expert) reviews and comments. As most scientists know, good reviews are time-consuming, that is, come at a cost. Even though community service out of idealism is a common trait for instance in the Wikipedia community, incentive systems can improve the situation for contributors as well as for the contributed content. They aim at reimbursing the review or revision cost by awards, and at invigorating the review process.

Privacy-friendly incentives complement this fundamental goal with anonymity and privacy protection for all users. Therefore, they enable a double-blind peer review process and nurture fairness, impartiality, and rigor. Authors as well as the reviewers of documents can remain anonymous during the entire review process. Such a blind review process is believed to be essential for high (academic) quality and honest comments, even though it sometimes lacks in reviewer accountability.

Our goal is to establish a cryptographic system that reaches high quality standards, while fulfilling the diverse requirements of the involved parties.

We formalize the incentive system as a collaborative document editing system, in which all revisions, reviews and comments are linked to one initial document P_0. We consider a document version history $\mathbb{P} = \{P_0, \dots P_n\}$ as ordered sequence of revisions, reviews and comments associated with the P_0, where P_n denotes the most recent revision or review.

Principals

There are multiple parties interacting with a document P. We have a clearing house that hosts all documents and organizes the incentive system, in our case the wiki W component. The wiki has a community of users and each user U may act in different and multiple roles:

Reader U: A reader consumes a document P. Any reader may offer incentives to other users to improve the quality of a document by a review or a revision.

Author V: An author contributes an initial version or a revision of a document P.

Reviewer R: A reviewer contributes reviews and comments for a document P in exchange for receiving an incentive.

Editor E: An editor is a privileged user, who may approve or decline document revisions or reviews by authors and reviewers.

We introduce a bank B to exchange electronic incentives for real-world goods and awards. Users of wiki W can withdraw fresh incentive e-coins and deposit spent ones as part of our virtual incentive economy. Even though we allow a system with full anonymity, we require each user to register with a trusted identity issuer I to infuse accountability in the entire review and incentive process. Each user U obtains an identity certificate σ_{U} on its identity sk_{U} from issuer I. Our system works with multiple banks as well as multiple identity issuers, we focus on the single-bank/single-issuer case for simplicity. The identity of an honest user is never revealed by the incentive system, whereas the certified identity enforces separation of duties between authors and reviewers, and prevents double-spending attacks as well as vandalism.

Concepts

In a privacy-friendly incentive system, many anonymous users interact with a single document P. Incentives may be given before or after a contribution (revision or review). *Pre-contribution* incentives are offered to users to provide a contribution at all and it is independent from the contribution quality. For instance, a reader U can offer incentive e-coins for any reviewer R who is willing to contribute a review. *Post-contribution* incentives are offered after the contribution is made and may be dependent on the quality of the contribution. For instance, users can rate the quality of reviewer's contribution and offer reputation e-coins for her work.

In our model, a reader U explicitly withdraws incentives from a bank B. The reader U offers these *pre-contribution* incentives on the wiki W for improvements on a document P. The wiki W acts as a clearing house and it is responsible for ensuring unlinkability by exchanging the spent incentives of reader U with bank B for fresh incentives. Once a reviewer R decides to contribute a review P', she submits the review to the wiki W for inspection by an editor E. Once the editor E approves the review, the reviewer R can obtain the incentives from the wiki W. As *post-contribution* incentives extension, the number of obtained incentives can be dependent on the review rating or the reviewer can obtain separate reputation e-coins to build a reputation credential.

Checks and balances

The privacy-friendly incentive system provides anonymity to all users and balances this property with strong accountability safe-guards. In a fully anonymous system without such safe-guards, malicious users could attempt to manipulate reviews, sabotage other author's work or publish fabrications without accountability. Well known examples of checks and balances to counter those attacks are the separation of reviewer and author/editor, or the binding of reviews and documents to the contributor's true identity.

To achieve accountability as well as separation of duties between roles, we introduce a cryptographic domain pseudonym $N_{P,\mathsf{U}}$ for each user U that interacts with a document P. It is a function of the user's true identity sk_{U} and the document P while hiding sk_{U} computationally. Therefore, each entity interacting with document P has one unique pseudonym, which is independent from entity's role. Pseudonyms $N_{P,\mathsf{U}}$ and $N_{Q,\mathsf{U}}$ created for different documents P and Q are unlinkable.

3.4.3 Author Reputation System and Trust Evaluation of Content in MediaWiki

Architecture

MediaWiki[5], the software used by Wikipedia, is probably the most used wiki-software. Therefore, the implementation of an author reputation system was done as an extension for this application. In the the following we outline how two of the core mechanisms from Section 3.2.2 can be implemented for the wiki scenario from Section 3.2.1. The requirements and design for this prototype are also published as a result of PrimeLife in [KPS11].

User reputations and certifications: For the credibility of authors, an *author reputation system* assigns *author reputation* to authors. This is done initially by using certifications users got outside the system (e.g., a master degree to show expertise in computer science) and transferring them to a reputation value in the author reputation system. Our reputation system allows to set up different fields of expertise and users can have different pseudonyms and different reputations in these fields. We make use of the identity management system developed by the PRIME project[6] (PRIME) for assisting the user in showing pseudonyms and certifications. For showing reputation PRIME was extended. After registering a user's reputation is influenced by the ratings other users give to the content she creates within the wiki system. Our reputation system uses natural numbers (including 0) as set of possible reputation values. As users manage their reputation on their own, one is able to omit single ratings. To avoid that users omit negative values, our system

[5] http://www.mediawiki.org/wiki/MediaWiki

[6] www.prime-project.eu

uses only positive ratings. Consequentially it does not make any sense for the user to omit any value.

Evaluating trustworthiness: A *content rating system* allows readers of content to judge on it's quality and give a *rating* to it. The content rating systems collects all ratings given, aggregates them to a reputation of the content and shows it together with the content to possible future readers. The rating a user gives to the content influences the aggregation algorithm depending on the reputation the rater shows about herself. The problem with wikis is that information changes frequently. The reputation extension is derived from the ReaderFeedback extension for MediaWiki.[7] Using a wiki as implementation platform brought in additional issues like several authors of a content and that there exist different versions of content that do not all get rated. Our content rating system makes use of 5 stars as possible ratings a content might get.

This means that our overall system consists of the following parts:

- the user-side with the PRIME version allowing for reputations installed,
- PRIME certification authorities for issuing credentials/certifications,
- the wiki server with the PrimeLife-ReaderFeedback-extension.

Functionality

In the following we describe the basic functionality of the system:

Fetching Initial Reputation. Authors may start work with an initial reputation. That means, that proofs of competence certified by an authorized institution can be brought in the work with the wiki by using certifications that have been given to the user. This is done by showing anonymous credentials with PRIME to the wiki server. From this, a certain reputation value an author has is calculated by the wiki.

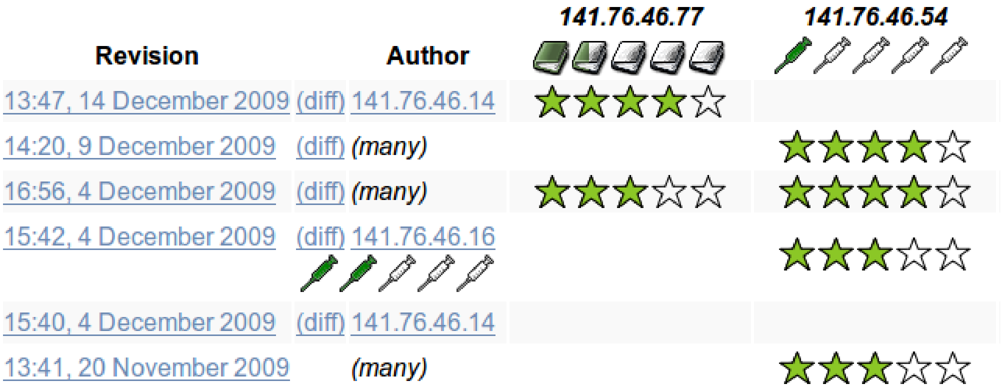
Passive Usage. When browsing a wiki page, which has been rated with the help of reputation extension, the user will see the reputation of the content of this page in form of one to five stars.

The reputation shown may not be the reputation calculated for to the latest revision of a page. This is due to the fact, that there may be no ratings given to the latest revision which is necessary for the calculation of the reputation value. However, if no ratings have been given to the latest revision, the most recent computable reputation value will be displayed. If a user wants to know more details of a reputation value the history of single ratings from which the reputation value was calculated can be shown as well (Figure 20).

The tabular representation contains much information in one view. The raters reputation is shown on top of the table below the name or IP address[8] of the rater. The

[7] http://www.mediawiki.org/wiki/Extension:ReaderFeedback

[8] If the user has a MediaWiki account the user's name is displayed otherwise the IP address is shown.



Revision		Author	141.76.46.77	141.76.46.54
			●○○○○	●○○○○
13:47, 14 December 2009	(diff)	141.76.46.14	★★★★☆	
14:20, 9 December 2009	(diff)	(many)		★★★★☆
16:56, 4 December 2009	(diff)	(many)	★★★☆☆	★★★★☆
15:42, 4 December 2009	(diff)	141.76.46.16 ●●○○○		★★★☆☆
15:40, 4 December 2009	(diff)	141.76.46.14		
13:41, 20 November 2009		(many)		★★★☆☆

Figure 20: Interface showing the Reputation and Rating History. The different icons represent the reputation type shown (e.g., the syringes represent a certain reputation in medical area).

different icons represent the type of reputation which was shown (e.g., the syringes represent a certain reputation in medical field). The stars below the raters are the ratings, which were given to a single revision of the page. If an author indicated her reputation together with submitting an edited page, this reputation is shown beneath the authors name or IP address.

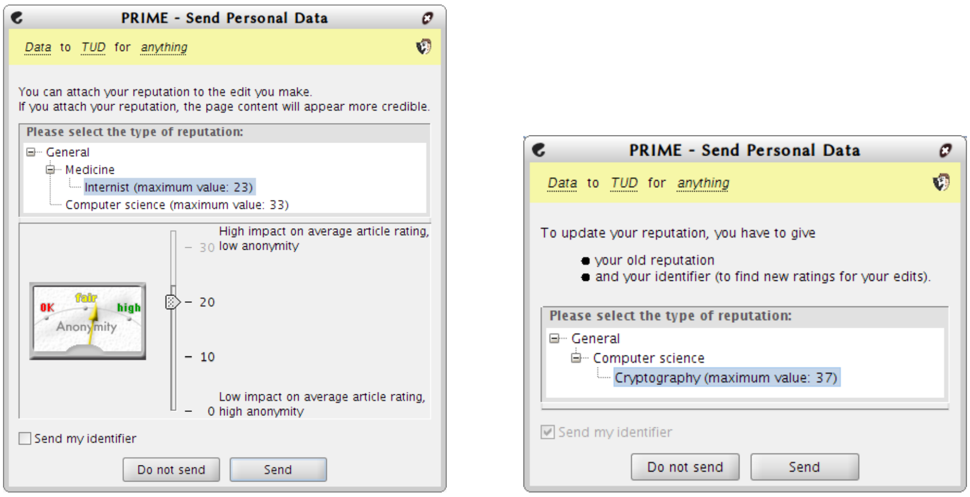
Editing Wiki Pages. When editing a page, a user is asked if she wants to send her reputation value. This reputation value is needed to calculate the reputation of the page afterwards. The higher the reputation value of the author is, the more impact it will have on the reputation value of the page. For showing reputation a user shows a credential. We make use of credentials that allow greater-than-proofs to allow an author to decide about the amount of reputation she reveals depending on her wish for anonymity. e.g., when having a reputation value of 63, an author may prove that she has a value greater than 20 or greater than 50. The higher a reputation value is, the more impact it will have on the reputation value of the page but as the set of authors shrinks when increasing the reputation value, the anonymity-set of the author shrinks as well.

As every user has to decide on this trade-off on her own, a so called “Send Personal Data Dialogue” asks the user for her reputation value and tries to display the trade-off in a graphical way. This dialogue is shown in Figure 21a.

Additionally, the type of the reputation is important for the calculation. While the topic of a page does not change, the author may have several reputation credentials. e.g., a surgeon may edit a page, the content of which is about gynecology. The reputation credential on surgery may have more impact on the gynecology article than a reputation credential dedicated to dentistry. However, having a credential on gynecology would have the most impact. An author may not only show a credential from her concrete reputation type. Within the issuing process she obtains a more general value automatically (e.g., while issuing a credential about gynecology, one obtains a credential about medicine and a general reputation credential as well). When asked for her credential, the user may decide if she shows the specific credential (which has more impact on the page-reputation) or if she uses the more general one (with the benefit, that the anonymity set is higher).

In addition to her reputation value and type, the user may send some identifier. This gives her the possibility to benefit w.r.t. the increase of her reputation value, whenever other raters give a high rating to the page. However, giving an identifier makes the user linkable of course. The decision about sending the identifier is done with a checkbox shown in Figure 21a on the bottom.

The identifier has to be shown again, when the user wants to fetch a reputation-update later. Figure 21b shows this dialogue.



PRIME - Send Personal Data

Data to TUD for anything

You can attach your reputation to the edit you make.
If you attach your reputation, the page content will appear more credible.

Please select the type of reputation:

- General
  - Medicine
    - Internist (maximum value: 23)
  - Computer science (maximum value: 33)

High impact on average article rating, low anonymity

Low impact on average article rating, high anonymity

□ Send my identifier

(a) Editing a page

PRIME - Send Personal Data

Data to TUD for anything

To update your reputation, you have to give

- your old reputation
- and your identifier (to find new ratings for your edits).

Please select the type of reputation:

- General
  - Computer science
    - Cryptography (maximum value: 37)

☑ Send my identifier

(b) Updating the Reputation

Figure 21: Customized Send Personal Data dialogue.

Rating Pages. In addition to editing, users can actively influence the reputation of a page value by rating it. Therefore, a rating form is shown to the user on the bottom of each page. With this form a user can give 1-5 stars in four criteria of a page (currently reliability, completeness, neutrality, presentation).

Similar to editing, the user is asked for her reputation value, when she wants to submit her rating. A dialogue similar to the one shown in Figure 21a is shown to the user when she wants to submit her rating. As also stated in the last section, several properties of the reputation value of the rater influences the impact of the page reputation as well as the anonymity of the rater. Therefore, raters have the same choice authors have between a large anonymity set or a high impact of her rating to the reputation value of the page.

Lessons learned

The prototype built provides a comprehensive framework for creating author reputation and content evaluation in a wiki by considering the complex and partly contradicting requirements such a system has. This concept could be applied also to other applications or cross applications, especially all applications that are PRIME-enabled. However, our first user evaluation has shown that the overall framework is too complicated for end users to understand intuitively and make use of all features, especially the privacy features.

3.5 Conclusive Remarks

The need for establishing trust in online content is obvious. We could show in our experiments, that the kind of meta information users require to make a trust assessment differs and that many users are not aware of which indicators they actually use. Nevertheless, according to our second experiment, users experience privacy concerns about their user reputation when actively contributing to the trustworthiness of online content by giving ratings to content.

We presented a set of functional mechanisms with requirements which help to establish trust in content, namely 'User reputation and certification,' 'Binding Metadata to Data,' and 'Evaluating trustworthiness'. Additionally we gave 'privacy-friendly incentives' as supportive means with requirements. These mechanisms were implemented as prototypes for two investigated scenarios, either wikis or blogs.

Chapter 4

Identity and Privacy Issues Throughout Life

4.1 Challenges and Requirements

Much research and development has been done during the past couple of years to assist users in managing their partial identities in the digital world by several types of identity management [BMH05]. A comprehensive privacy-enhancing identity management system would include the following components [CK01]:

- an Identity Manager (IdM) on the user's side,
- IdM support in applications (e.g., at content providers, web shops, etc.),
- various third-party services (e.g., certification authorities, identity providers).

However, current concepts for identity management systems implicitly focus on the present (including the near future and recent past) only. The sensitivity of many identity attributes and the need to protect them throughout a human being's entire lifespan is currently not dealt with.

> *The digital lifespan is the range of time from the emergence of the first information that is related to the human being until the point in time when no more personal data is generated: from the moment of birth until death.*

Hence, lifespan refers to the temporal aspects of privacy and identity management and, in particular, to the challenges related to realising (privacy-related) protection goals over very long periods of time. The area of challenges regarding privacy is vast – even when not considering an entire lifespan (see, e.g., [Eni08]). This chapter discusses additional problems of lifelong protection of individuals concerning their privacy in a technology-based society.

4.1.1 Dealing with Dynamics Having an Influence on Privacy Protection

New possibilities of data storage, faster processing, better data analyses, and entirely new means of data processing, such as those based on sensors, result in an intensified impact on and possible infringement of an individual's right to privacy. In recent years the miniaturisation of processors, ubiquitous computing, and remote access to and collection of data without the data subject's awareness have posed new threats to informational self-determination.

In the attempt to protect the privacy of information relating to her, an individual encounters several types of dynamics she has to deal with. On the one hand, there are the several stages of life she herself passes through. Typically, these are three stages, namely those of nonage, adulthood and old age (retirement). All of them have particular characteristics related to the individual's ability to manage her privacy. On the other hand, the individual's ability to manage her private sphere is also influenced by technological developments, administrative developments, and the evolution of society itself.

Stages of life

The aforementioned stages of life to a large extent determine the individual's ability and willingness to manage her private sphere. This is not only due to the individual's ability to deal with the technical devices necessary but also by the ability to understand the privacy and security-relevant aspects concerning her private sphere.

> *A stage of life of an individual with respect to handling her privacy is a period in her life in which her ability to manage her private sphere remains between defined boundaries characterising this stage of life.*

The ability of a baby to deal with the privacy aspects of her personal data is obviously non-existent. When a child reaches adolescence, her ability to interact with the technical means to safeguard her personal sphere reaches a stage, where she can independently react appropriately to privacy infringements. In this period, the parents will have to step back and relinquish responsibility for managing the personal sphere of their children, as it is the case with many other areas of life. Once the adolescent has become a mature person, she holds the responsibility to take care of the safeguarding of the privacy aspects of her personal information for all intents and purposes. When reaching the stage of old age, a similar process takes place but then in an opposite direction. Figure 22 illustrates the individual's possibility of managing her private sphere during the various stages of life.

This development is not usually taking place in such a straight line. Adults may have temporary or permanent needs, that others support them or even act on their behalf concerning decisions on their private sphere. This becomes more evident in the stage of old age. For small children as well as for very old people and in the case of emergency, delegation of the right to manage one's private sphere is needed. For children, these delegates automatically are their parents, in case of emergency or for old people it might be a close relative. Intermittently, it is not uncommon that individuals,

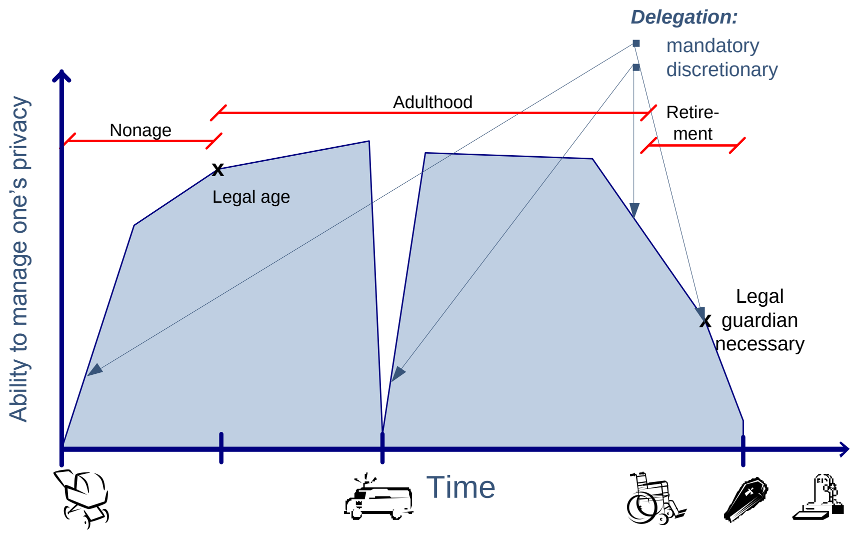


Figure 22: Stages of life: variation in the ability to manage one's privacy.

who in principle are able to manage their privacy on their own, want to involve other parties or delegate their privacy control.

A privacy-enhancing identity management system should support the delegation of duties and authorities. There are three possible situations that might occur regarding delegation from the legal perspective:

1. Delegation might be made by law automatically for a certain time frame (e.g., for children to their parents).

2. Delegation might be made willingly by an individual to others for a certain time frame (e.g., delivering mail to others during holidays).

3. Delegation of an individual might be initiated by other individuals to achieve delegation of her rights to them or others (e.g., in the case of incapacitating a person), which presumably requires thorough juridical investigation before divesting the person of a right.

Delegation can be implemented by different means. Usually, the delegate does not take over the identity of the individual concerned, but receives authorisations to act – often within defined ranges – on behalf of or as inheritor, respectively. Technical processes for delegation and digital estate have to be defined in accordance with legal procedures. We come back to this issue in Section 4.1.3. and should allow for:

- granting delegations by a person herself, automatically, or by others,
- revoking delegations by a person herself, automatically, or by others,
- accountability and relief of delegates,
- showing or not showing to others whether a person herself or her delegate acts.

It is necessary that all parties involved are aware of the legal and technical procedures, the conditions under which they may take place, and also possible liability issues. For accountability reasons, the activities of a delegate often would have to be logged so that in the case of dispute the performed actions can be analysed. This again may affect the delegate's private sphere.

Increasing disclosure of personal data within different areas of life

In line with the development of technological advances, such as Internet and Web 2.0, the reasonable expectation of privacy is affected and adapted. One may argue, that the code, in the sense of software and hardware, used in the networked society, suffers from a privacy myopia: Developers are not aware of the privacy risks their increased use of personal data cause. The increasingly sophisticated technological measures applied progress accordingly at the same pace as the increasing disclosure of personal data.

From the point of view of the stages of life the individual passes through, the disclosure of her personal data as used by these technological measures shows a similar progressive development. This is only natural because the individual becomes increasingly of interest to data processors but also because during her active life her data track grows, i.e. her digital footprint assumes larger proportions (see Section 4.1.2). The individual needs to use the services offered by data processors because disclosure and processing of data is officially required (e.g. because of school attendance, tax liability, or the obligation to contact the registry office when changing one's residence), or because the data are needed to fulfill tasks in the areas of e-commerce, leisure, communication etc.

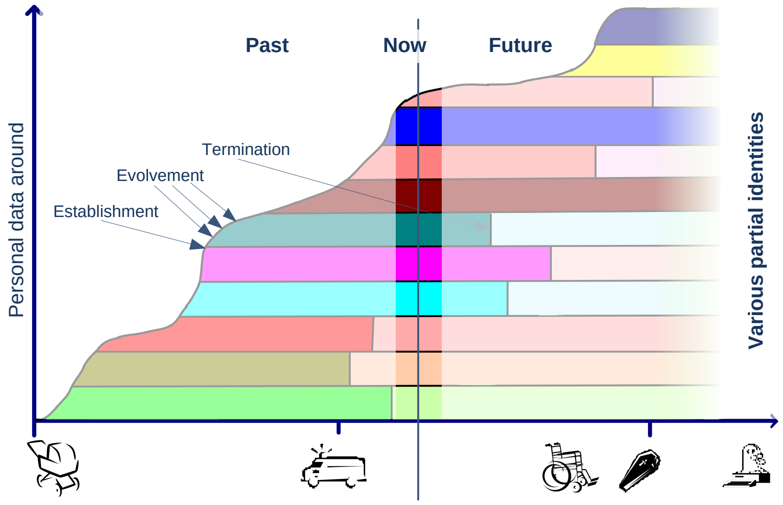


Figure 23: Accumulation of personal data and formation of partial identities.

Figure 23 shows a simplified model of increasing data disclosure to different data controllers, depicted by coloured stripes. The lighter colours on the right-hand side of data controllers express that the personal data are not needed anymore for the task to be fulfilled, but the data may still live on in official archives, at Internet services, or in the storage of communication partners . Note that the data are neither automatically deleted after the time of death nor after the funeral.

The coloured stripes shown in Figure 23 also correspond to several *partial identities* [HPS08], for example (but not exclusively) individuals in different areas of their life.

> *Areas of life are sufficiently distinct domains of social interactions that fulfil a particular purpose (for the data subject) or function (for society).*

Formal areas of life include education, work, and health care. Informal areas of life cover mainly a user's social network including family, friends, leisure, religion etc. Some of these informal areas might become formal by institutionalisation, e.g., for religion, in the form of membership in a church.

Technological considerations

Lifelong privacy mechanisms need to cover not only the near past and future of an individual, but also need to consider the future prospects for a human's lifetime and also beyond, which means about 100 years and more. The problem is that we only have experience with computer-based technology for less than half of this time, and experience of exchange of computer data over the Internet for less than a quarter of this time. This means that all privacy mechanisms (including public-key cryptography invented in 1976 based on cryptographic assumptions) could not be tested in practice for a whole lifetime yet. For the selection of privacy technology (hardware and software), attention should be paid to the following aspects:

- The duration of cryptographic security (based on cryptographic assumptions) should be at least an individual's lifetime. If unconditional security (not relying on cryptographic assumptions, but just on pure probability theory) is possible and feasible, it should be provided.

- Migration of data between different hardware and software needs to be assured. When considering the long-term risks in an unpredictable setting, the sensitivity of personal data is of utmost importance. For the choice and protection level of personal data processed, a categorisation regarding their sensitivity with respect to privacy has to be made [HPS08, CHP^{+}09].

4.1.2 Digital Footprint

A major challenge related to privacy and identity management in current society lies in the digitization of information. The use of computers, laptops, smart phones and all sorts of other devices has not only changed the format in which data are stored, but also the amount and types of information that are generated and collected. Individuals consciously create numerous partial identities to perform actions on the Web. Unlike in traditional, analogue, communications, data storage requires little space and the data can easily be copied, transferred, or processed otherwise. In addition, each interaction with a device can generate data about that interaction (metadata), such as date and time, type of device, location, and what data were processed. Interaction between devices or via the Internet leads to several places where data are generated and stored, implying that, usually, once the data are out there, they will stay forever. The Internet does not forget. The accumulated data an individual leaves behind in numerous interactions via or with digital devices can be called a Digital Footprint. Just like a footprint in the sand, digital interactions leave traces that can indicate the individual's (earlier) presence.

> *Digital footprints are data that accumulate in information systems and can be ascertained as belonging to one individual. This means that data in this sense is not restricted to personal data only.*

This section describes how digital footprints evolve and how extensive they can be. With regard to unique identifiers, there will be a brief overview of how governments facilitate connections between data sets by issuing these identifiers and using them for different purposes.

Lifelong footprints

The above-mentioned characteristics clearly indicate some challenges with respect to privacy and identity management throughout life. Data are easily copied and transferred, so it is difficult to have a clear view on where data are stored or otherwise processed and by whom. The fact that the data remain available for a long time adds further problems with regard to the lifespan of the individual. While an individual's status, interests, preferences, and so on may be subject to changes over time, data relating to these aspects may remain static. The lack of (automatical) updating data can lead to discrepancies when data sets are compared. Besides, the data may remain even after an individual has deceased, which results in a digital lifespan that can be much longer than the actual lifespan of the individual. And also prior to birth, data relating to the unborn child are collected and stored in several databases. The actual lifespan of the individual and the digital lifespan of this individual, thus, do not exactly overlap. From a strictly legal perspective, this may be problematic, since data protection legislation only applies to data relating to a data subject. A data subject is defined as a 'natural person', which is explained as a living human being. So, data processed before birth or after death do basically not fall under the scope of data protection legislation.

The traces left behind in digital interactions, including the metadata, bring up another challenge. The metadata are often collected or created by the party with whom an individual interacts or by third parties that are in principle not involved in the interaction. For instance, advertising companies track and trace the web behavior of individuals in order to present personalized or targeted advertisements, therewith increasing revenues. The individual visits a website and does not consciously or on purpose communicate with this advertising company. However, by analyzing the web behavior, the advertising companies try to derive interests and preferences of the individual; to create a personal profile. This profile can be seen as an (partial) identity of the individual, but is created by a third party without the knowledge of the individual herself. In terms of identity management, this causes major problems. An individual is unable to choose what characteristics to disclose and, more importantly, the profiling can reveal characteristics the individual herself was not even aware of or disagrees with. Several aspects, such as the creation of context-specific partial identities and informational self-determination are at stake. From the perspective of privacy and identity management, these aspects are essential in order to enable the individual to maintain control over her identity.

Linking Data

When taking into account that each and every transaction in an information system leaves a digital trace behind, it becomes clear that the accumulation of data can be enormous and that it thus becomes easier to link data to a single individual. The more data available, the more unique the combinations of these data are. The more data disclosed to a party over time, the lower the fingerprinting threshold becomes, which means that after a certain amount of time individuals can be uniquely identified and recognised when they appear in a new transaction, even when this new transaction is done from another computer or another location than previous ones [Con09].

The possibilities of linking data that belong to one individual also have their drawbacks on the dynamics in the surroundings of the individual. At first glance, it might seem that changing surroundings can have a positive influence on identity management, because contexts can be separated relatively easily. Nevertheless, linkability becomes possible when disclosing information or revealing certain patterns over time, meaning that the different surroundings can be connected as belonging to the same individual. These issues are worsened by the fact that the Internet does not forget. There is no digital oblivion. Once data are revealed somewhere, they persist over time and remain stored in databases, cache memories and so on and so forth.

In order to link data sets, some kind of connector is needed. Sophisticated technologies, such as clicking behavior, click trails, and key strokes can be used, but often there is a common identifier (e.g. IP address, registration number) that allows for linkage. The easiest way to link data sets is by the use of a so-called unique identifier, which is an identifier that identifies a given individual. Unique identification can be supported by governments who issue unique identification numbers to their citizens. These numbers are sometimes even used throughout different domains or areas of life, thereby linking these areas and the data therein as belonging to one individual. In the Netherlands a unique identifier, the BSN (Burger Service Number, Citizen Service Number), is commonly used in several settings which, in principle, allows for the construction of a compound identity, instead of the citizen having distinct identities in different areas. The BSN can be used in online interactions and is printed on physical documents, such as passports, driving licenses, and health insurance cards. Other means are the use of national ID cards, as is the case in, for instance, Germany, France, and Austria. Belgium, Sweden, Ireland, and Poland do use unique identification numbers, often combined with ID cards.

Sensitive attributes

Some attributes and attribute values usually need more privacy protection than others. According to [HPS08, CHP^{+}09], we distinguish the following properties of identity attributes, which, alone or in combination, pose specific risks to privacy when being disclosed:

- *Data may be static, or changes are quite accurately predictable:* Data which are static over time and are disclosed in different situations enable linkage of related data. Examples for static data are date and place of birth. Similar to static data are those which are quite accurately predictable or guessable because they follow some rules. Examples are data following mathematical rules like the number of children that will typically only remain or increase. If static identity information is being used for purposes such as authentication, this bears a risk because these data cannot easily be revoked and substituted: For example, the use of fingerprints with biometric access systems.

- *Data may be (initially) determined by others:* Data that the individual concerned cannot determine herself (e.g., the first name) may persist or it may take a significant amount of time or great effort to change them. A special case is the inheritance of properties from others, e.g., the DNA being inherited from the natural parents.

- *Change of data by oneself may be impossible or hard to achieve:* If data are static (see above) or if data are not under the individual's control, wilful changes may not be possible. Examples are data processed by an organisation.

- *Inclusion of non-detachable information:* There are data that cannot be disclosed without simultaneously also disclosing some side information tied to the data. Examples are simple sequence numbers for identity cards, which often reveal gender, birth data and at least a rough timeframe of when the identity card was issued.

- *Singularising:* If data enable the recognition of an individual within a larger group of individuals, the individual may be tracked or located, even if other personal data of the individual are kept private.

- *Prone to discrimination or social sorting:* There are no data that are definitely resistant against possible discrimination forever. This does not need the individual to be identified or singularised. If some people disclose a property and others resist to do so, this already allows for social sorting or positive discrimination.

Note that this list of sensitive properties extends the enumeration of special categories from Art. 8 Data Protection Directive ("personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, trade-union membership, and the processing of data concerning health or sex life"). Because of the sensitivity of the listed personal data, everybody should be careful with related data processing.

4.1.3 Concepts for Delegation

User-controlled identity management systems have been proposed as a means to manage one's own private sphere. Such systems support the data subject in managing her privacy and identity needs, e.g., to determine whether to enter a contract depending on the privacy policy of the data controller or which partial identity to use when communicating with others.

In certain phases of life, the data subject cannot manage her privacy and identity needs by herself, thus she needs more support than an identity management system alone can offer. This includes situations in which the data subject is not able to manage her privacy and identity needs for a limited time or forever as well as those situations in which the data subject is not yet able to manage her needs. Here, a delegate can be appointed who should act in the interest of the person concerned.

Delegation can be defined as transfer of legal representation power from one natural person to another natural person. This transfer can either result from provisions, which lay down legal prerequisites, or from the concerned natural person's decision [WP109b]. This idea has been described in PrimeLife Heartbeats H1.3.5 [WP109b] and H.1.3.7 [WP110c] on which the WP1.3 demonstrator illustrated in H1.3.6 [WP110d] is based. The underlying concepts for delegation - also in connection with aspects of lifelong privacy management - will be shown in this section.

Privacy-relevant challenges concerning delegation

Different privacy-relevant challenges have to be met in order to implement a working delegation system into identity management systems. First, delegation requirements have to be analysed, which has already been done within the PrimeLife project [WP109b]:

- Data controllers should foresee that data subjects can delegate their identity management to proxies.
- Data controllers should enable delegation of identity management limited to specific proxies and specific scopes (such as purposes, application, data controllers, time etc.).
- Data controllers should enable revocation of delegation of identity management under defined conditions.
- Data controllers should provide mechanisms for a data subject to get an overview of decisions by her proxy regarding the processing of personal data.
- Data controllers should provide concepts and mechanisms for identity management after one's death.

These requirements have to be examined from various aspects: delegation based on legal provisions and delegation based on explicit consent of the data subject.

1. Delegation based on legal provisions. During certain phases of life, the data subject needs to be represented by another natural person. This may start when a child is born and it may continue in case of adults that may have temporary or permanent needs to get support, and it may finally end with the death of the data subject or her last will [WP109b].

The right of informational self-determination is a fundamental right. Fundamental law does not explicitly allow for representation by others. Fundamental rights are by nature non-transferable, personal rights. It is uncertain what impact this might have for the exercising of the data subject's right of informational self-determination. Taking the German civil law as an example, representation for transactions is possible according to section 164 ff. BGB (Bürgerliches Gesetzbuch, German Civil Code). Legal regulations for representation can also be found in data protection laws, e.g., section 28 paragraph 1 BDSG (Bundesdatenschutzgesetz, German Federal Data Protection Act). This legal provision regulates that the proxy has to process personal data of the individual represented in case of a contract. Thus, legal representation can impact fundamental rights as a secondary effect.

2. Delegation based on explicit consent. Various reasons exist why a data subject may wish to transfer the full or partial legal authority of representation to another individual. Assumed the data subject uses an identity management system, the legal basis for delegating rights would usually be the consent of the data subject. Delivering the contractual duties, however, will possibly also require the processing of personal data, especially the data necessary to fulfil the actual goal of data processing [WP109b]. Therefore, delegation based on explicit consent will typically be the legal basis for a delegate using an identity management system.

The concrete delegation requirements below result from the requirements above:

- Data controllers should provide mechanisms for issuance of the mandate of the proxy, invocation of actions under the name of the principal with the mandate, verification of the mandate, revocation of the mandate from the proxy and expression of acceptance of the mandate by the proxy.
- Data controllers should support derived credentials for proxies that enable the proxy to use own credentials to get access and act on behalf of the principal.
- Data controllers should provide mechanisms that allow the principal to trace actions taken by the proxy.
- Data controllers should provide mechanisms for the principle to declare preferences and conditions to the power of the proxy.
- Data controllers should provide mechanisms to maintain the proxy's private sphere.
- Data controllers should define how to deal with the data subject's data after her death. Introducing solutions to meet these requirements into the WP 1.3 demonstrator was one of the challenges within the development of the demonstrator.

Delegation at different stages of life

As described above, delegation can be an important means to cope with changing abilities of privacy management during the different stages of life of a data subject. The need of being represented in managing one's privacy usually starts with the data subject's birth or even before. In the prenatal phase ("Fruit of the womb") [HRSZ10], personal data might be processed in prenatal DNA tests. Enabling the parents to manage even this data supported by an identity management system could thus be helpful.

But not only the unborn data subject, but also children and teenagers need representatives to manage their privacy needs. Children and teenagers have legal representatives until they reach the legal age. Due to their aging and thus growing autonomy, their right to decide for themselves grows, in transactions as well as in their privacy rights. Thus, regulations have to be implemented allowing the child to trace which decisions their parents made for them. Besides, mechanisms have to be implemented to enable the grown-up child to revoke decisions once made by their parents concerning the child's privacy needs.

But delegation can also have an important means for adults lacking privacy management capacities. If the data subject is seriously ill or not present for a certain period of time, she won't be able to exercise her identity needs as usual. Delegating these rights of acting on behalf of the data subject to a predefined person for a predefined scenario enables the data subject to have her privacy needs managed even in these situations. If the delegate has access rights to the identity management system, managing the data subject's privacy needs will not be too complicated for the delegate and the data subject will be enabled to understand which transactions happened while her absence from using the system.

Managing privacy needs may also be an important means for deceased people ("... to tomb") [HRSZ10]. When an individual dies, the instrument of law succession applies. Although Article 1 of the Data Protection Directive 95/46 EC assigns the right of privacy and data protection to natural (living) persons only, in some European legal frameworks a "post-mortal personality right" is provided [WP109b]. Especially users of Social Network Sites (SNS) disclose various personal data while using the SNS. After the data subject's death, these data usually won't be deleted automatically. Predefining a delegate who exercises the data subject's rights after her death enables the data subject to ensure the kind of data handling she desires. Besides, it enables the SNS provider to short-term react on the delegate's actions after the data subject's death.

Recommendation for implementing privacy-aware delegation

The approach to implement delegation in the PrimeLife Backup Demonstrator has already been analysed in PrimeLife Heartbeat H1.3.7 [WP110c]. The results additional to the above can be summarised as follows:

1. Limiting the delegate's access to the necessary extent. Therefore the demonstrator should support the delegator, i.e., the data subject, in the following steps[1]:

- How to generate delegation requests to delegate candidates?
- How to deal with their (positive/negative/missing) answers to those requests?
- How to revoke the status of being a delegate?
- How to limit the access rights of the delegate?

[1] A comprehensive list of descriptions of the entities indicated here is given in Section 4.2.

- How to communicate possible conditions to being a delegate or conditions to having the actual "power of authority"?
- How to communicate to the delegate that she is assigned the actual "power of authority"?

The question of the actual "power of authority" that a delegate should have if she acts on behalf of the primary user is tackled in [WP110d] by issuance of a "credential verifying a certain status of the primary user". For visualising the functionality of the demonstrator, this construction may be sufficient [WP110c].

2. Controlling the delegate's actions.

- The system has to ensure that each transaction of the delegate will be logged and that the data subject will be enabled to revoke delegate's transactions. This way of controlling the delegate's actions can also be called "Ex-post user control" [WP110c]. Therefore it builds on the legal requirement of the right to rectify, erase or block the data (Article 12 of the Data Protection Directive 95/46/EC).
- The system has to ensure that the delegate can only act in predefined scenarios and only for the predefined period of time (see above).
- The system has to ensure that the delegate uses different access keys than the data subject to facilitate traceability and transparency for the data subject.
- The system has to ensure that logged data will only be stored as long as necessary. Additional, it has to ensure that the delegate (and the data subject) is informed about logged accesses and storage mechanisms and periods.

Implementing the above into the demonstrator will be the basis for a working delegation system.

Summary

The implementation of the requirements described above into the WP1.3 demonstrator will be analysed and evaluated in detail in PrimeLife Heartbeat H1.3.8. A first short summary of the evaluation and therefore also of the evaluation of implemented delegation options will be given below in Section 4.2.3.

4.2 Demonstrator

A scenario that is relevant to all areas and stages of life and deals with dynamics and possible delegation – this means the challenges we described in the previous Sections 4.1.1 and 4.1.3 – is the area of backup and synchronisation tools and applications.

Many backup systems and backup strategies, which have been available for many years, are already dealing with the problem of unwanted data loss. However, they are mostly protecting the raw data only and do not involve the data subject, her specific characteristics, social relations and interactions as a part of their scope. Existing backup systems and backup strategies also do not reflect the process of evolution of the data subject during her lifetime with respect to the possible different states she might pass through during her lifetime and which might have an immense influence on her ability to manage her data on her own behalf (e.g., illness, hospitalisation, or death). Additionally, existing systems and strategies dealing with the problem of unwanted data loss do not cope with boundaries among distinct areas of the data subject's social interactions. However, these aspects are nowadays becoming more and more sensible on the level of the data, hand in hand with the massive expansion of the technology.

Therefore, we decided to analyse the problem of unwanted data loss from the perspective of lifelong privacy. Current solutions do not provide a sufficient level of data protection with respect to lifelong privacy management of the data subject holding the data. Thus, we decided to demonstrate that it is possible to solve the problems induced by the requirements on lifelong privacy when protecting the data subject against unwanted data loss.

The proposed privacy-enhanced backup and synchronisation demonstrator (referred to as the PrimeLife Backup Demonstrator) focuses on the following problems related to lifelong privacy:

1. Protection of the data subject against unwanted data loss during her lifetime by redundancy and physical distribution of the data.

   The problem of unwanted data loss can be solved by redundancy and the physical distribution of multiple copies of the data provided by backup and synchronisation tools. In the PrimeLife Backup Demonstrator, we are proposing to solve the problem of unwanted data loss by taking advantage of services provided by online storage providers which are nowadays available on the Internet (for example Dropbox, Apple MobileMe, Windows Live SkyDrive, Ubuntu One and others) and store multiple copies of the data in a distributed environment. Distribution of potentially sensitive backup data in such kind of environment, however, leads to confidentiality problems.

2. Assurance of lifelong confidentiality of the data subject's data stored in a distributed environment.

   The problem of data confidentiality in a distributed and untrusted environment can be solved by the encryption of the data. Encryption must assure that only the authorised data subject (whom the data belongs to) is able to operate with her data stored in the distributed backups by default. Nobody else should have implicit access to the data even after the death of the data subject. On the other hand, during the lifetime of the data subject, unpredictable situations might occur, which might temporarily or permanently limit her in her ability to access her own data (for instance in case of her illness, hospitalisation or death). This might lead to situations that her data, which might be important for other parties relying on it (possibly in a legal relationship with the data subject), is not accessible by these parties when needed (for example important work documents) or is permanently lost.

3. Delegation of access rights to the data subject's backup data allowing other parties to operate with her data if specific conditions are fulfilled.

Delegation capability of the PrimeLife Backup Demonstrator allows other parties authorised by the data subject (whom the data belongs to) to access her backup data in case particular conditions specified by the data subject are satisfied. Delegation of access rights of the data subject's backup data could in general lead to situations where authorised parties with corresponding access rights are not only able to access the desired data but also other data possibly covering other areas of the data subject's life, which they are not authorised to access. This might, however, not be desired by the data subject herself.

4. Distribution of the backup data according to different areas of life of the data subject and her different partial identities.

   Distribution of the backup data according to particular areas of the data subject's life or her different partial identities enables the data subject to manage her privacy in such a way that allows her to physically and logically separate her data related to distinct domains of her social interaction.

Besides the above mentioned problems, additional non-trivial issues must be addressed, which are covered by the high-level requirements on prototypes developed within the PrimeLife project (cf. [WP111]). As far as the PrimeLife Backup Demonstrator is based on the backup and synchronisation functionality, it also has to address further privacy-related issues amplified by the backup and synchronisation nature (cf. [WP110d]).

In order to understand the descriptions in the following sections, it is helpful to be familiar with the following terminology:

Terms:

Primary item: is an original item for which one or more backup items are created during the back up action. In a general sense, a primary item can be referred to as any determinated set of data, which has one or more copies called backup items dedicated for backup purposes. A primary item can be a file but it can also be a more specific type of data such as, for instance, an e-mail, a contact, or even settings on the TV.

Backup item: is a copy of a primary item stored in a backup. A backup item reflects the data of a primary item at the time when the backup item was created. Note that even if each backup item must belong to one and only one primary item, this primary item may not exist during the entire lifetime of the backup item. A backup item can exist in several versions at a particular point of time.

Backup: is a non-empty set of backup items.

Backup task: describes which specific set of primary items should be backed up to which storage provider according to which schedule. The output of a run of a given backup task is a backup.

Actors:

Primary user: data subject who owns/holds primary items.

Storage provider: provides storage space for backups.

Delegate: is an entity that receives particular rights on the backup from a delegator.

Delegator: is an entity that has the privilege to delegate rights to delegates concerning a particular backup. In most applications of the PrimeLife Backup Demonstrator, the primary user acts as the delegator.

Delegate candidate: is an entity that was selected by a delegator to act as a delegate but does not possess particular rights yet.

Delegation request: represents a request sent to the delegate candidate asking her whether she accepts particular rights from the delegator.

Credential issuer: is an entity that issues a credential verifying a certain status of the primary user. This status can for example be: "primary user is ill," "primary user is hospitalised," "primary user is dead," or others.

4.2.1 Overview of the PrimeLife Backup Demonstrator Architecture

After describing the overall goals and visions with respect to the backup scenario in the previous section, we will report on the current state of the design and implementation of the actual PrimeLife Backup Demonstrator in the rest of this chapter.

The PrimeLife Backup Demonstrator consists of three main components[2]:

1. **the core**, which offers the main functionality. The core is written in Java and runs as a background process on the machine, which holds the data (primary items) that should be backed up. The core makes its functionality accessible using a REST[3]-like interface.

2. **a Web-based user interface** (called "backup console"), which is written using HTML, CSS, JavaScript and Ajax. It can be displayed using an ordinary web browser. It utilises the REST calls offered by the core to accomplish the tasks desired by the user.

3. **a tray icon** shown in the notification area of the taskbar found in many modern operating systems. This tray icon informs the user about important information and status changes related to the backup process. Moreover, it allows the user to launch the backup console.

Basic building blocks used by the core

The core is the central place that provides all the methods necessary to create backup tasks and actual backups, to restore them, to manage delegations etc. Moreover, it manages all the data (configuration information, credentials etc.) related to this.

[2] For closer look at the structure of the PrimeLife Backup Demonstrator please refer to [**?**].

[3] REST – Representational State Transfer

The entire functionality is provided through REST-like calls. Thereby HTTP is used as the underlying application level protocol. Therefore, the core contains an embedded web server (namely Jetty[4]). The binding between the HTTP URLs and the Java methods is done with the help of the “Java API for RESTful Web Service” (JAX-RS[5]). JAX-RS uses Java annotations, which simplifies the process of making a Java method available as web service. Particularly the Jersey[6] reference implementation of JAX-RS is used.

In case the URL itself does not encode all parameters and data necessary for a given REST call, the HTTP body contains the additional data needed as input for the given REST call. The data transmitted in the HTTP body can be encoded either using XML or JSON[7]. The desired encoding type is set by the client. The marshalling/unmarshalling is done using JAXB[8] (in case of XML encoding) and Jackson[9] (in case of JSON encoding). Both APIs allow an automatic marshalling/unmarshalling of the data, thus avoiding any need for manually implementing serialisation methods for every data type used.

File system and external storage providers

When it comes to the question of where to store the backups, the PrimeLife Backup Demonstrator follows the trend to store data “online,” e.g., by using dedicated online storage providers or more generally spoken “in the cloud.” Although storing backup data more “locally” is supported by the PrimeLife Backup Demonstrator (e.g., on an external hard drive connected to the machine), using online storage is the more important use case. The reason for this is clearly not the aspect of “following trends”. It is rather driven by the “lifelong” aspects which should be demonstrated.

On the one hand, the backup scenario implies that the data (backups) are available for the entire life of the primary user. Clearly managing a large set of external hard drives would lead to much more burden on the user compared to managing contracts with online storage providers. Moreover, by using online storage, the data is accessible from every place in the world where an Internet connection is available. This makes the whole process of delegating access rights to a backup much more feasible. Finally, it is easier to store the backups as truly redundant copies, which in turn is a precondition for avoiding long-term data losses.

On the other hand, using online storage leads to interesting research questions with respect to the “lifelong privacy” aspects. First of all, the backed up data needs to be stored in a way so that only authorised entities can gain access to the actual backup content. Besides this need to achieve confidentiality of the backed up data, the “privacy” of the involved entities (e.g., primary user, delegates etc.) should be assured as well. In our concept, it means that each online storage provider should learn as little as possible about the involved parties (e.g., who accesses which backup, at which time, how often etc.).

[4] http://eclipse.org/jetty/
[5] http://jcp.org/en/jsr/detail?id=311
[6] https://jersey.dev.java.net/
[7] JSON – JavaScript Object Notation
[8] http://jcp.org/en/jsr/detail?id=222
[9] http://jackson.codehaus.org/

The PrimeLife Backup Demonstrator uses the "Apache Commons Virtual File System"[10] library, which allows access to various different local and remote file systems by a single API. Besides the online storage providers, which are supported by default (like SFTP, WebDAV, FTP etc.), plug-ins for well-known online storage providers (like Dropbox) were developed. Although these plug-ins were deployed together with the software of the PrimeLife Backup Demonstrator, conceptually they could be downloaded from the Internet as well. The general idea is that either some kind of directory service lists available online storage providers and provides the necessary plug-ins or that at least a given online storage provider offers a plug-in for the service on her web site.

A storage provider plug-in would not only implement all the necessary functionality to actually access the online storage but it will also provide user interface components which allow a user to create a new account with this storage provider. This in turn comprises e.g., the necessary SLA[11] and the payment.

With respect to the trustworthiness (or more general, the properties) of an online storage provider, the prime assumption is related to availability, i.e., it is assumed that some data stored at a given storage provider would be (with high probability) available according to the negotiated SLA. Beyond that, each storage provider is seen as a potential attacker (in the sense of the concepts of multi-lateral security). Especially, it should not be necessary to trust the storage provider with respect to confidentiality or integrity of the stored data. Nor should the storage provider be trusted with respect to the privacy of a given user. Therefore the PrimeLife Backup Demonstrator needs to implement appropriated measures to achieve these protection goals (e.g., by means of cryptographic mechanisms like encryption, integrity protection etc.).

In order to reduce the linkability between different transactions done by the same user with a given online storage provider (or multiple storage providers), it is assumed that a communication layer anonymisation service is used. If the access to the online storage is based on HTTP (like WebDAV, Dropbox etc.), existing anonymisation services like AN.ON[12] or Tor[13] could be used.

Nevertheless, the linkability usually remains at the application layer. Because we want to support existing (legacy) online storage providers, we cannot assume that they base their access control on unlinkable anonymous credentials. Rather, the common combination of login/password would be used. In this case, the only way to avoid linkability, e.g., that two backups belong to the same user, a user has to create multiple accounts (ideally using multiple online storage providers). Note that a special privacy-enhanced storage provider could be built in the future incorporating anonymous credentials for access control. More concrete, the implementation of this storage provider could be based on the Identity Mixer[14] anonymous credentials, which is a part of the PRIME Core[15] developed within the EU FP6 integrated project "Prime"[16] and the PrimeLife project.

[10] http://commons.apache.org/vfs/

[11] SLA – Service Level Agreement

[12] http://anon.inf.tu-dresden.de/

[13] https://www.torproject.org/

[14] http://www.primelife.eu/results/opensource/55-identity-mixer/

[15] http://www.primelife.eu/results/opensource/73-prime-core/

[16] https://www.prime-project.eu/

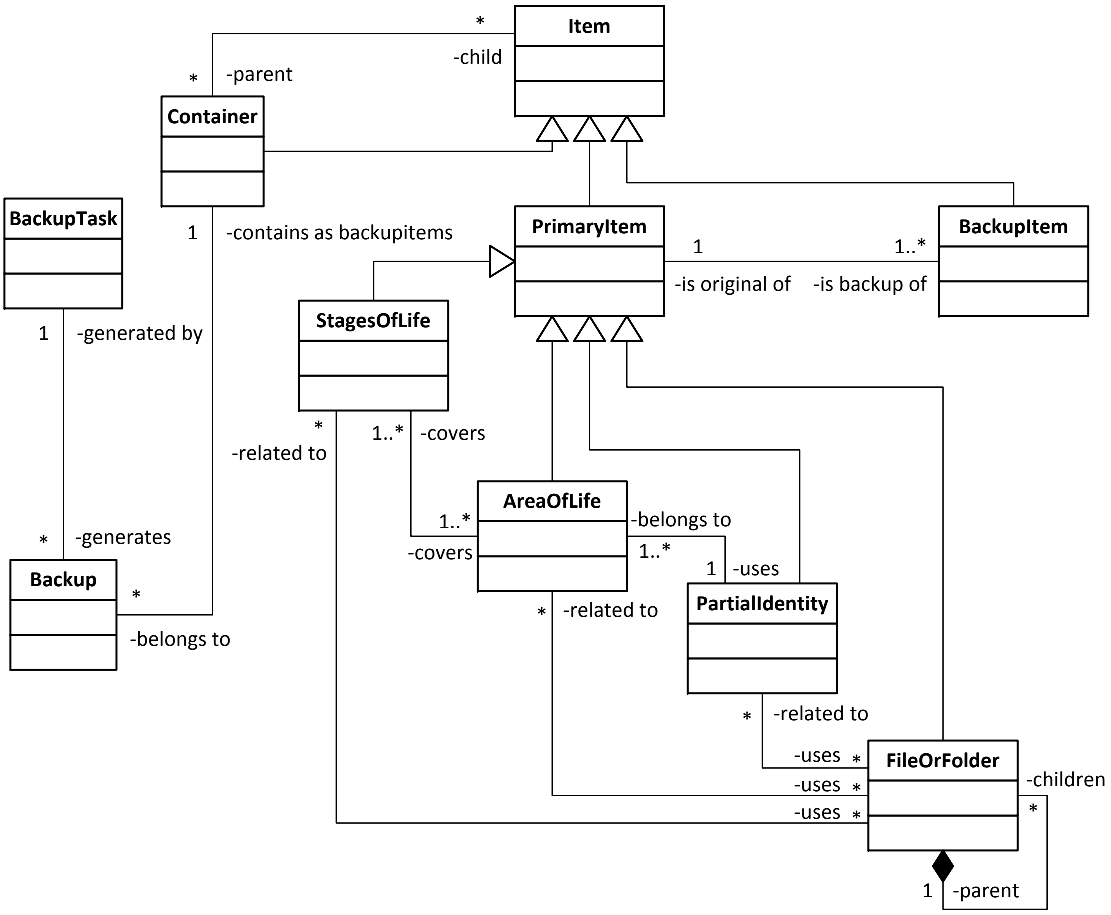


Figure 24: Relations among of Areas of Life, partial identities, files etc.

Backup and Restore

"Backup" and "Restore" are the most prominent functionalities of existing backup solution. Moreover, most backup tools operate on a *data* level, that is the user selects files, directories, partitions or whole disk drives. This kind of selection mode is supported in the PrimeLife Backup Demonstrator as well.

In addition to this common data level-based selection mechanisms, the PrimeLife Backup Demonstrator offers an *identity*-based selection mode. Remember that a basic concept of privacy-enhanced identity management is the separation into different partial identities, areas of life and stages of life. Thus, the user can select from each of these categories, e.g., specify which area(s) of life or partial identities she wants to backup.

Items of different types can be grouped together within a so-called container. A container can be understood as template of the primary items to be backed up. This would ease the related selection during the creation of a new backup task.

Areas of life, partial identities and files are related to each other (see Figure 24). An area of life typically covers multiple partial identities; likewise a partial identity is related to many files. Note that a file can be related to more than one partial identity and area of life, e.g., a digital photo that shows persons from the "Family Area of Life" as well as the “Working Area of Life.”

In the field of privacy-enhanced identity management, one of the basic reasons for differentiating between different partial identities, areas of life etc. is to avoid unwanted linkability between them. Consequently, this unlinkability property has to be preserved for the backups as well. Thus, even if the user can group different partial identities or areas of life together for creating one logical backup task, the actual backup data need to be stored separately[17]. Assume for instance, that the user selects files that belong to two different partial identities. In this case, two different backups will be created, ideally stored on two different online storage providers. Otherwise an attacker might learn that the two partial identities in question actually belong to one and the same user.

The PrimeLife Backup Demonstrator is assumed to interplay with a privacy-enhanced identity management system, which provides information about the existing areas of life, partial identities and the relation among them and the files stored on the given machine to the PrimeLife Backup Demonstrator. Since such an IDM system does not really exist today in practice, we decided to create mockup data (areas of life, partial identities, files and folders) that are used for demonstrating the privacy-preserving backup aspects related to them. Though the PrimeLife Backup Demonstrator allows for creating backups of real files and folders, these items are not associated with any partial identity or area of life. From a privacy (linkability) point of view, they are treated as being “equal” and therefore are stored within a single backup.

Delegation

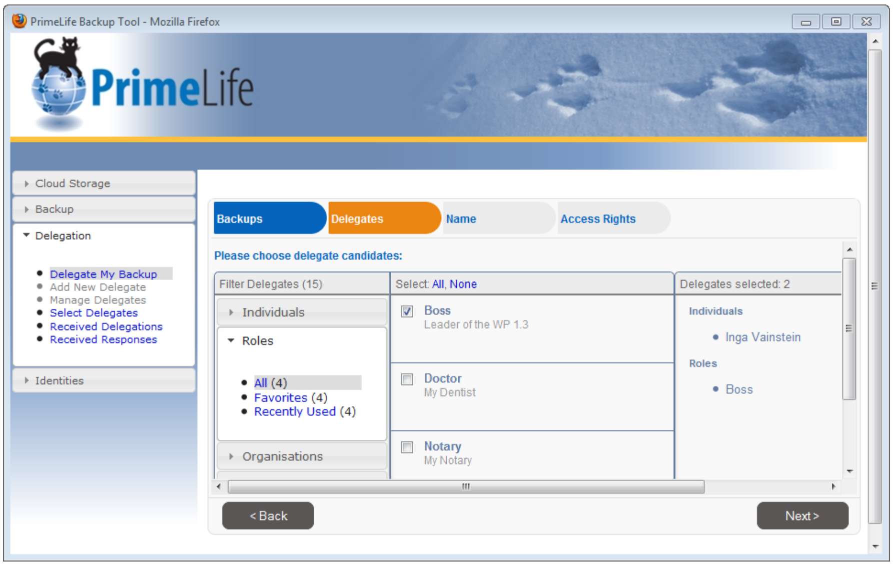
Delegation is one of the most important aspects to be shown by the PrimeLife Backup Demonstrator. In addition to being an innovative feature in itself, it also has many implications with respect to the area of lifelong privacy, which in turn is the underlying motivation for the PrimeLife Backup Demonstrator.

From a functional point of view, delegation means that a primary user (the delegator) delegates access rights to a particular backup to some delegates (see Figure 25). These access rights are usually bound to a policy describing under which circumstances the access should be granted. A typical use case would be that a user delegates the access rights to the files related to her work partial identity/area of life to her colleagues under the policy that access is permitted only if the user becomes ill. Moreover, the policy would express the obligation that the user is informed if one of her colleagues really accesses the backup.

Delegation does not only deal with the privacy of the delegator but also with the privacy of the delegates. Therefore, a delegate will be asked if she is willing to accept the delegation request. Thereby – in the spirit of informed consents – she will be informed that each access to the backup would also be reported to the delegator.

It is currently an open question as to how much additional “meta-data” about a given backup will be communicated to a delegate. On the one hand, a delegate might want to know beforehand which files are included in a given backup, so that she can better decide if she really wants to accept the delegation or access the backup, respectively. On the other hand, this information might already have some negative impact on the

[17] This is currently not handled in the current version of the PrimeLife Backup Demonstrator.



Please choose delegate candidates:

☑ Boss
Leader of the WP 1.3

□ Doctor
My Dentist

□ Notary
My Notary

Delegates selected: 2
Individuals
- Inga Vainstein

Roles
- Boss

Figure 25: Management console of the PrimeLife Backup Demonstrator showing the interface for selection of delegates.

privacy of the delegator. For enhancing the privacy of the delegator, it is desirable that a delegate only learns that information if she actually accesses a given backup.[18] Thus, if the conditions defined in the access policy never become true, unnecessary information flow will be avoided.

In the current version of the PrimeLife Backup Demonstrator, the list of possible delegates is predefined as mockup data. There are plans to integrate other sources of information for these address book-like data. Possible sources are social networks such as Facebook, Xing, LinkedIn etc. Moreover, the current version of the PrimeLife Backup Demonstrator uses plain e-mail messages for transmitting delegations and the related acceptance responses. Future versions of the PrimeLife Backup Demonstrator might use the communication infrastructure offered by the mentioned social networks as well. In that case we will of course use techniques like Scramble! to protect the confidentiality of the delegation.

The delegation itself is an XML document describing the contents of the backup, the location of the backup and under which circumstances (policy) the backup can be accessed by the delegate. The delegation might also transfer credentials (e.g., issued by the delegator) to the delegate, which the delegate will need in order to access the backup.

Conceptually, a lot of modern cryptographic mechanisms exist that could be used to construct a privacy-enhanced protocol for the purpose of delegation. Such a solution would require infrastructural support that is not in place today. Examples would be anonymous credentials issued by governmental or public institutions, public key infrastructures, attribute-based access control mechanisms etc. Therefore we decided to

[18] Note that “access a given backup” does not necessarily mean, that she has to extract all the data stored within a given backup but could just mean that she accesses the meta-data (e.g. list of files) in order to decide which data she actually wants to restore.

implement a much simpler scenario, which on the one hand is much closer to the current practice and which, on the other hand, integrates components developed within the PrimeLife project and thus would not only demonstrate delegation itself, but also illustrate the interplay of the various components developed within the PrimeLife project.

Our delegation scenario comprises the following entities/components:

1. The eCV, a component which allows a user to store an electronic version of her Curriculum Vitae (CV). This component was developed within the PrimeLife project to demonstrate privacy aspects in service-oriented architectures (see [WP611]).
2. A trusted third party (TTP) utilising the PrimeLife policy engine (see [WP510]).
3. A legacy online storage provider.
4. The delegator.
5. A delegate.

In this scenario, we demonstrate how a delegator can delegate the access to her backup to a delegate who can access the backup in case the delegator is ill. The delegation then would comprise the following steps:

1. The delegator stores the encrypted backup at the legacy online storage provider. The encryption is done using a symmetric cipher and the key k.
2. The delegator generates a random value k_1 and calculates k_2 such that $k = k_1 \oplus k_2$. Note that $k_2 = k \oplus k_1$, i.e., k_2 can be seen as a one time pad encryption of k using the key k_1.
3. The delegator stores k_1 at the TTP. A PPL[19] policy regulates the access to k_1 saying that:
   - Only give access within a certain time frame and
   - to someone who can present a credential C_1 and
   - a credential proving that the delegator is currently ill.
   - The obligation is to delete k_1 after the time frame is over and
   - to inform the delegator if someone has accessed k_1.
4. The delegator sends k_2, C_1, C_2 to the delegate (encrypted under the public key of the delegate). The delegator informs the delegate about the circumstances under which (illness of delegator and valid time frame) and how the delegate can access the backup.

Now let's assume that the delegator becomes ill. In this case, the delegator (or her doctor) sends a certification of illness to the eCV of the delegator.[20] Moreover, the

[19] PrimeLife Policy Language

[20] Note that such kind of infrastructure is already under development, e.g. Google Health (http://www.google.com/health/), Microsoft HealthVault (www.healthvault.com/) or the German eHealth card.

access policy to that certification says that someone who shows credential C_2 is allowed to access the certificate of illness (in this case the delegator should be informed by the eCV).

In case the delegate wants to access the backup of the delegator and thus uses the rights delegated to her, the following steps happen:

1. The delegate downloads the encrypted backup. How the access control to the encrypted backup is done depends on the methods provided by the legacy online storage provider. Usually there exist means of sharing the stored data with a defined set of users. But a broader discussion of this issue is out of scope here.

2. The delegate shows credential C_2 to the eCV and requests the certificate of illness of the delegator. Note that the delegate has received C_2 in step 4 during the delegation process.

3. The delegate requests k_1 from the TTP. she therefore shows credential C_1 together with the certificate of illness of the delegator to the TTP. The delegate gets k_1 and the TTP informs the delegator about that access to k_1.

4. Now the delegate is able to calculate $k = k_1 \oplus k_2$ and can decrypt the encrypted backup of the delegator.

The description above shows a very brief overview explaining the general ideas we have with respect to using existing components developed by the PrimeLife project for the PrimeLife Backup Demonstrator. Further research needs to be done to avoid/remove all the linkage that remains between the different steps (e.g., using different transaction pseudonyms for delegator and delegate etc.). Moreover, the information the various parties learn should be minimised (e.g., the TTP does not need to know that the "certificate of illness" is actually a certification of illness (e.g., we could use "meaningless" credentials here)). As a final remark, please note that all the parties mentioned above can be distributed (in the usual k out of n setting). To some extent, the involvement of the TTP already is a kind of distribution, because the eCV and the TTP can be seen as entities that store information accessible under a given access policy.

4.2.2 Deployment of the Demonstrator

Due to the platform independent design based on Java and web-technologies, the PrimeLife Backup Demonstrator can be installed and run on many modern operating systems including Linux, Mac OS X and Windows. Nevertheless, we decided to create a dedicated virtual machine based on VirtualBox[21] as virtual machine monitor and Ubuntu[22] Linux as guest operating system, which contains all the necessary components pre-installed. This makes the process of running the demonstrator much easier, especially if it comes to the more complex delegation scenarios. Besides the demonstrator itself, the virtual machine contains a local WebDAV online storage provider, two separate user accounts (one for the primary user/delegator and one for the delegate), and a local e-mail infrastructure (SMTP and IMAP servers). In order to make the installation of the virtual

[21] http://www.virtualbox.org/

[22] http://www.ubuntu.com/

machine itself as easy as possible, a screencast explaining the necessary steps was created (see http://prime.inf.tu-dresden.de/backupdemo/).

4.2.3 Lessons Learned from Evaluation of the Demonstrator

One of the focal points in evaluating the current status of the prototype was the comparison with the success criteria defined in the PrimeLife DoW. These requirements are:

- Requirements raised are complete,
- Prototype is functional and addresses identified requirements,
- Prototypes have been tested with real users.

In addition, further evaluation results will be described subsequently.

Requirements raised are complete

This success criteria was not part of the work on the demonstrator, nevertheless the demonstrator covers the essential set of those requirements that have been defined in different Heartbeats [WP109b], [WP110c]. In general it is hard if not impossible to prove completeness of a set of requirements, but the elaborated list is quite comprehensive and has been proven to work for many scenarios during the PrimeLife's lifetime.

Prototype is functional and addresses identified requirements

As described above, the prototype has to address the requirements identified. Essentially, the current implementation and additional conceptual documents meet these requirements, i.e. the solutions for implementing the defined requirements are functional and fit for practice.

The user-friendly handling of the backup system might serve as an example: Creating, managing, or restoring the backup system is very easy. Each step while creating one's backup is clearly illustrated, cf. Figure 26.

The same applies for the management of the primary user's backup, cf. Figure 27. The primary user is enabled to check out her backup management system as often as she wants to. The necessary information about her backup is always being provided.

Thus, one of the high-level requirements, namely the necessity for all parties involved to let the processing be controllable and controlled throughout the full lifecycle, is met by the current demonstrator. The user-friendly handling also grants transparency, one of the main basics of data protection, see inter alia Article 10 of the Data Protection Directive 95/46/EC [Dir95].

The delegation functionality is well designed, too. The primary user is enabled to delegate specific items or access rights of her backup items for specific purposes to specific delegates. Besides, limited time frames are provided by default for each delegation, cf. Figure 28. Thus, the demonstrator provides an overview about all important aspects of delegation at any time the primary user wants to. The system therefore also covers the necessity for the identity management system to define and make transparent who can

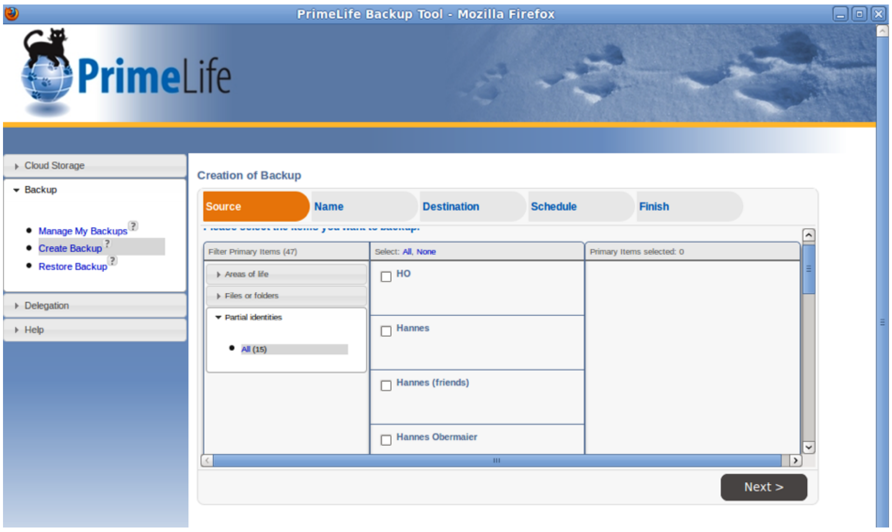
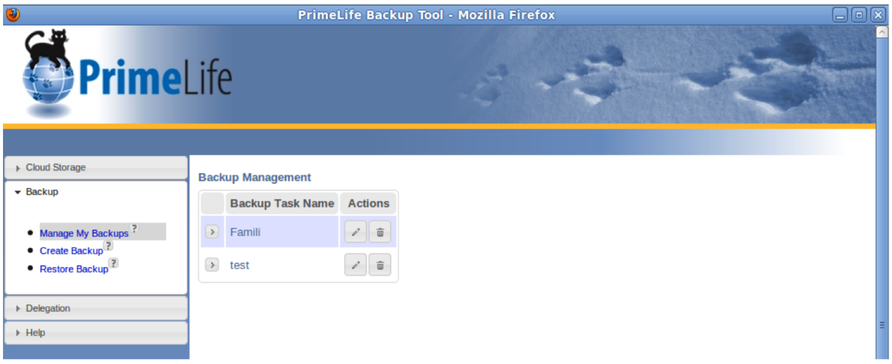


Figure 26: User interface for backup creation.



	Backup Task Name	Actions
›	Famili	
›	test	

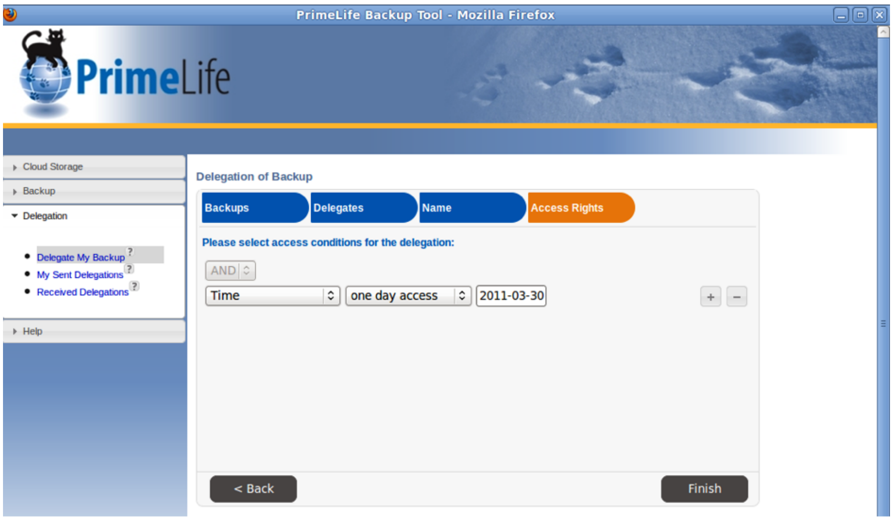
Figure 27: User interface for backup management.

get access to the log data under which circumstances. It also covers the necessity for the identity management system to foresee that primary users can delegate their identity management to delegates as well as the necessity for the identity management system to limit the primary user's consent in time by default.

As illustrated in Figure 29, different purposes for delegation and different access conditions can be chosen, regarding the different areas of life and various reasons for granting a mandate. The demonstrator offers defining access conditions among others for the following typical use cases:

- The primary user is ill.
- The primary user is hospitalised.
- The primary user is dead.

Another option for delegating access rights is the mandate just for one day without defining a special reason or purpose. Besides, individual purposes or use cases for



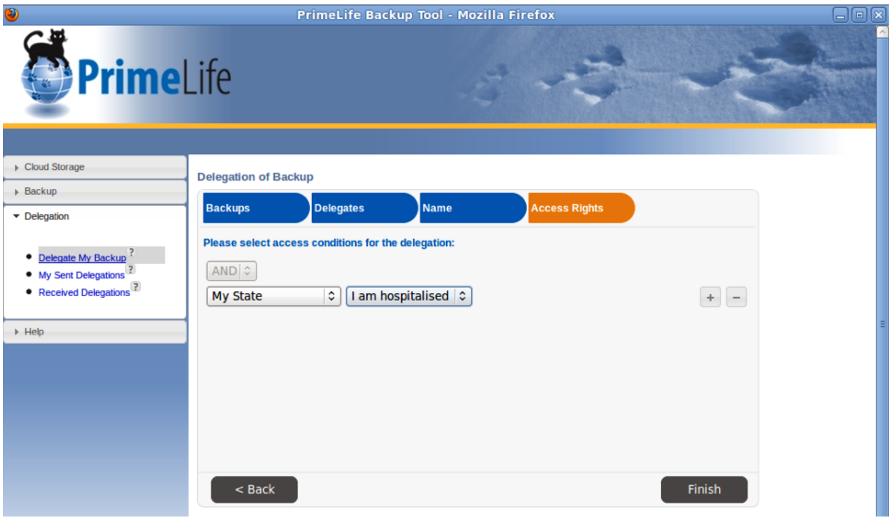
Delegation of Backup

Please select access conditions for the delegation:

Time | one day access | 2011-03-30

Figure 28: User interface for delegation time frames.

delegating access rights are conceivable. The option of giving a mandate just for one day might cover them as well as mandates for a longer term - for an individual use case to be defined by the primary user, cf. Figure 30. The demonstrator just offers predefined access conditions for living situations that might typically necessitate delegation.



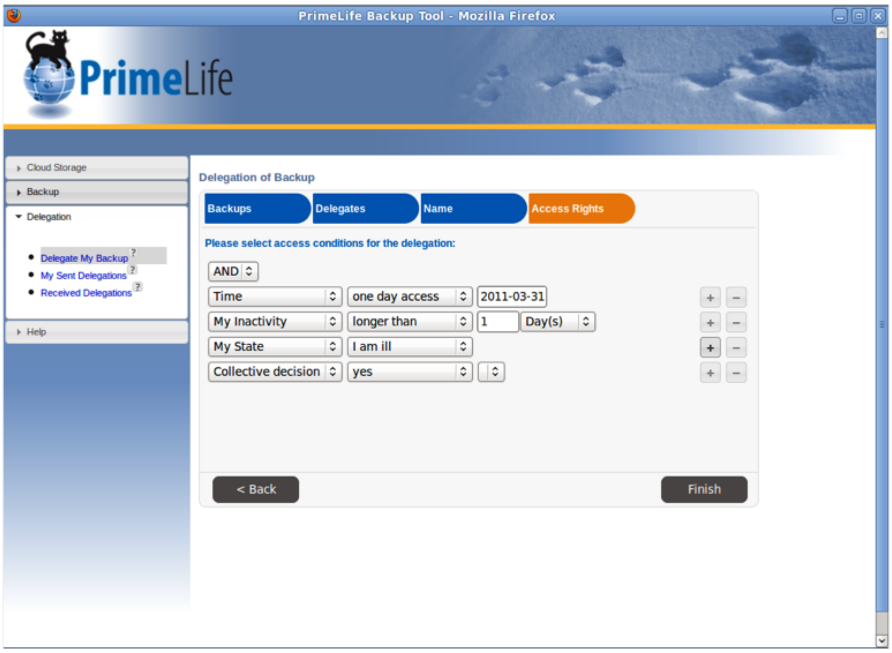
Delegation of Backup

Please select access conditions for the delegation:

My State | I am hospitalised

Figure 29: User interface for delegation purposes.

As one identity management functionality, the demonstrator supports the primary user in appointing multiple delegates for different partial identities or different scopes. Nevertheless, the primary user can also combine different delegation situations for one delegate - including access rights to the same data items. The query functionality available at each step of giving a mandate provides the information necessary for the primary user to have possible risks in mind while combining different delegations for different purposes in one delegate only. The user interface again provides this information in a



Delegation of Backup

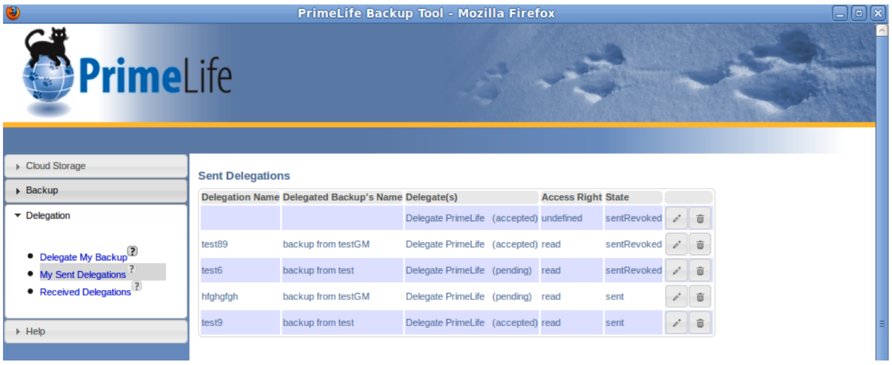
Please select access conditions for the delegation:

AND

Time	one day access	2011-03-31	
My Inactivity	longer than	1	Day(s)
My State	I am ill		
Collective decision	yes		

Figure 30: User interface for different access conditions.

comprehensible way.



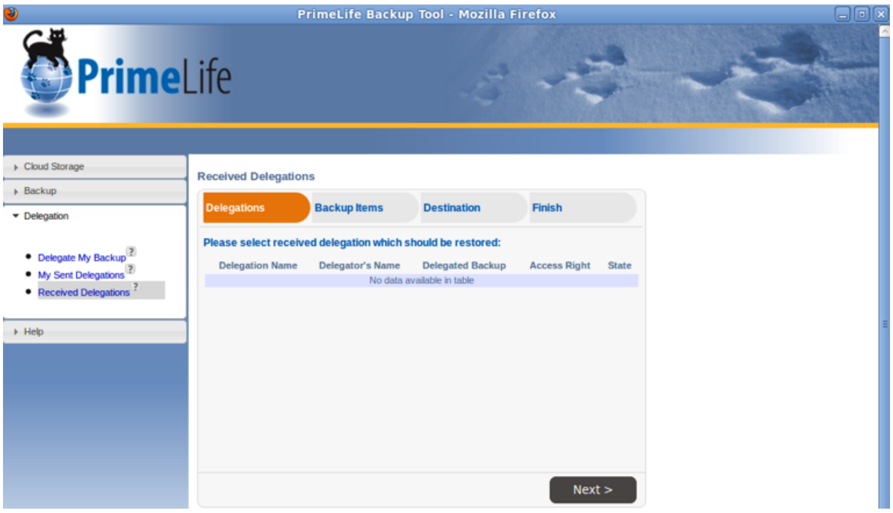
Sent Delegations

Delegation Name	Delegated Backup's Name	Delegate(s)	Access Right	State
		Delegate PrimeLife (accepted)	undefined	sentRevoked
test89	backup from testGM	Delegate PrimeLife (accepted)	read	sentRevoked
test6	backup from test	Delegate PrimeLife (pending)	read	sentRevoked
hfghgfgh	backup from testGM	Delegate PrimeLife (pending)	read	sent
test9	backup from test	Delegate PrimeLife (accepted)	read	sent

Figure 31: User interface for "sent delegations".

The demonstrator also provides information to the primary user on her appointed delegates and sent delegation requests. An overview can be seen under "My Sent Delegations", cf. Figure 31. This terminology was not immediately understood by all test users, so here another term could be used. This overview functionality grants transparency on the one hand and enables the primary user to check whether she has issued valid mandates for all possible use cases on the other hand.

In addition to the overview on issued mandates, the demonstrator provides information about "received delegations", i.e., delegation requests from a primary user to the current user, now in the role of a delegate. This again addresses the issue of transparency, cf. Figure 32. Although the term "received delegations" was easier to understand than the previously mentioned term "sent delegations", the vocabulary used by the demon-



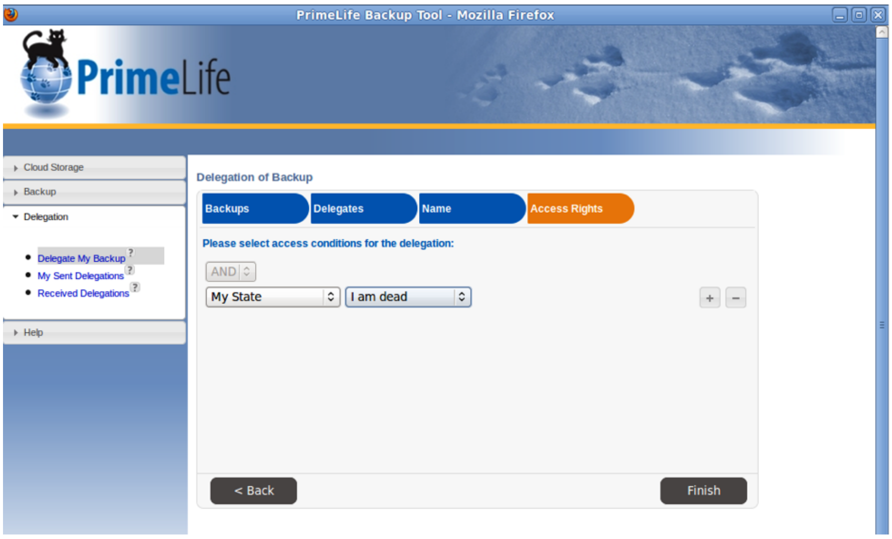
Received Delegations

Please select received delegation which should be restored:

Delegation Name	Delegator's Name	Delegated Backup	Access Right	State
No data available in table				

Figure 32: User interface for “received delegations”.

strator could be rethought.



Delegation of Backup

Please select access conditions for the delegation:

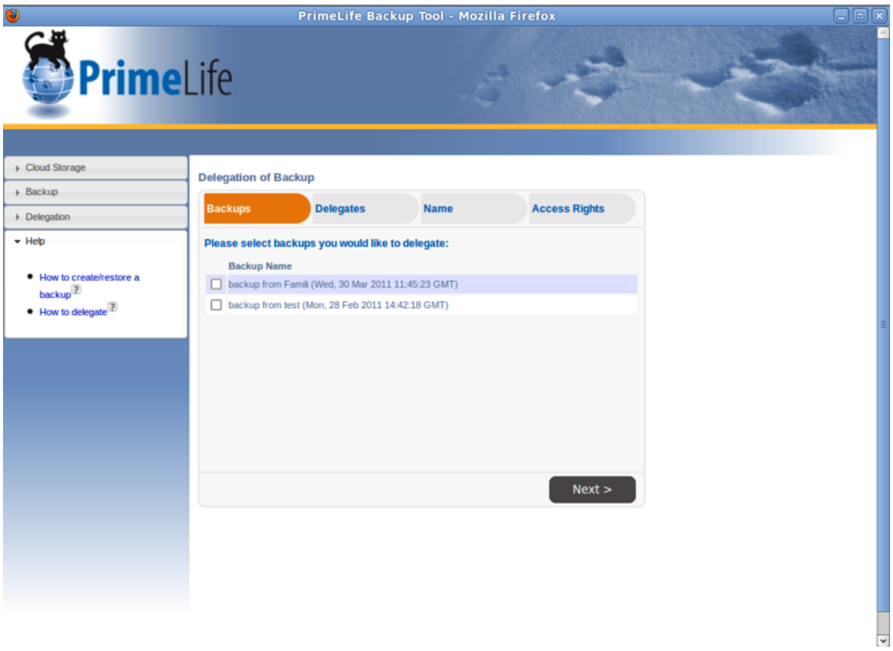
AND

My State | I am dead

Figure 33: User interface for data handling after the primary user’s death.

The demonstrator also provides delegation solutions for data handling after the primary user’s death. Thus, the primary user can avoid dissent between legal successors and the management system about handling her data after her death on the one hand and ensure effective data handling even beyond the grave at least by the appointed delegates. This solution covers the requirement of necessity for the identity management system to provide concepts and mechanisms for identity management after one’s death, covering all stages throughout the full lifetime. Figure 33 shows the possibility of choosing the state “I am dead”.

Clarity on data handling, storage and delegation is also provided by the help functionality within all the different functionalities of the demonstrator. The help function-



Delegation of Backup

Please select backups you would like to delegate:

	Backup Name
□	backup from Famili (Wed, 30 Mar 2011 11:45:23 GMT)
□	backup from test (Mon, 28 Feb 2011 14:42:18 GMT)

Figure 34: Help functionality within the user interface.

ality not only contains a manual on how to create a backup and how to delegate, but it also works in a context-dependent mode, giving the user the information on each usage option when working with the demonstrator. Thus, the primary user is enabled to think over her stored data and settings at any time.

This again addresses the necessity for all parties involved to have clarity under which circumstances decisions are revocable/irrevocable and what the potential impact can be. In particular, data controllers should inform primary users on to which degree their decisions are revocable or not and about the logic behind data processing in a comprehensible way.

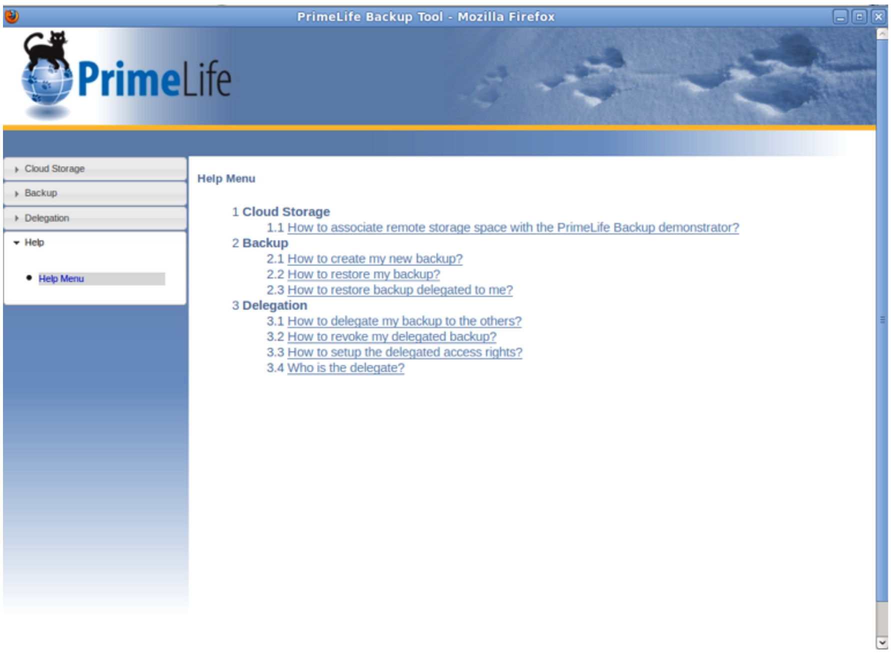
Besides, the necessity for the identity management system to define and make transparent who can get access to the log data under which circumstances is also addressed. Figure 34 illustrates how the user interface provides context-dependent help to the primary user.

The general help functionality also provides detailed information how to use the single items and options of the demonstrator. A guideline how to exercise the demonstrator's functionalities is included, too, cf. Figure 35.

Further, there is an own setting for the delegate after she has logged in into the backup management system. It offers inter alia the option of changing some of the data sets (only those she can access) to some extent or to add some personal information, cf. Figure 36.

Before a user logs into the system, she is provided with an overview about other users or delegates currently available in the system, cf. Figure 37.

Thus, transparency again is being provided. This is a first step in implementing the requirement to provide mechanisms that allow the primary user to trace actions taken by the delegate. If the primary user knows that a delegate is using her mandate, she



Help Menu

1 Cloud Storage
1.1 How to associate remote storage space with the PrimeLife Backup demonstrator?
2 Backup
2.1 How to create my new backup?
2.2 How to restore my backup?
2.3 How to restore backup delegated to me?
3 Delegation
3.1 How to delegate my backup to the others?
3.2 How to revoke my delegated backup?
3.3 How to setup the delegated access rights?
3.4 Who is the delegate?

Figure 35: Detailed information within the help functionality.

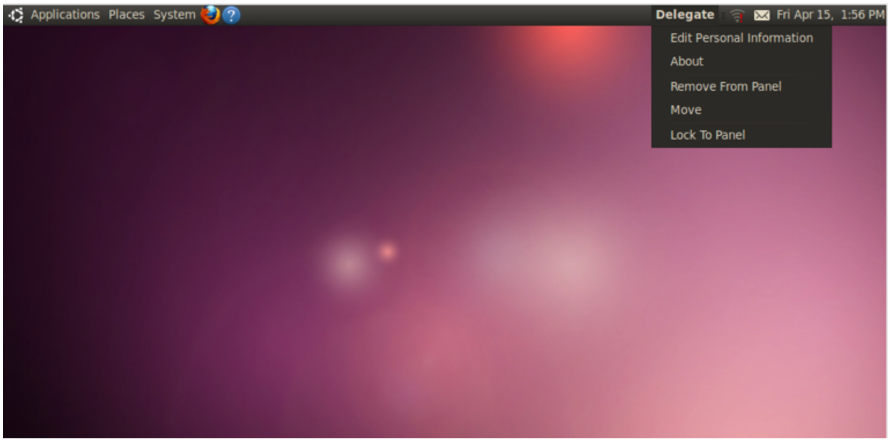
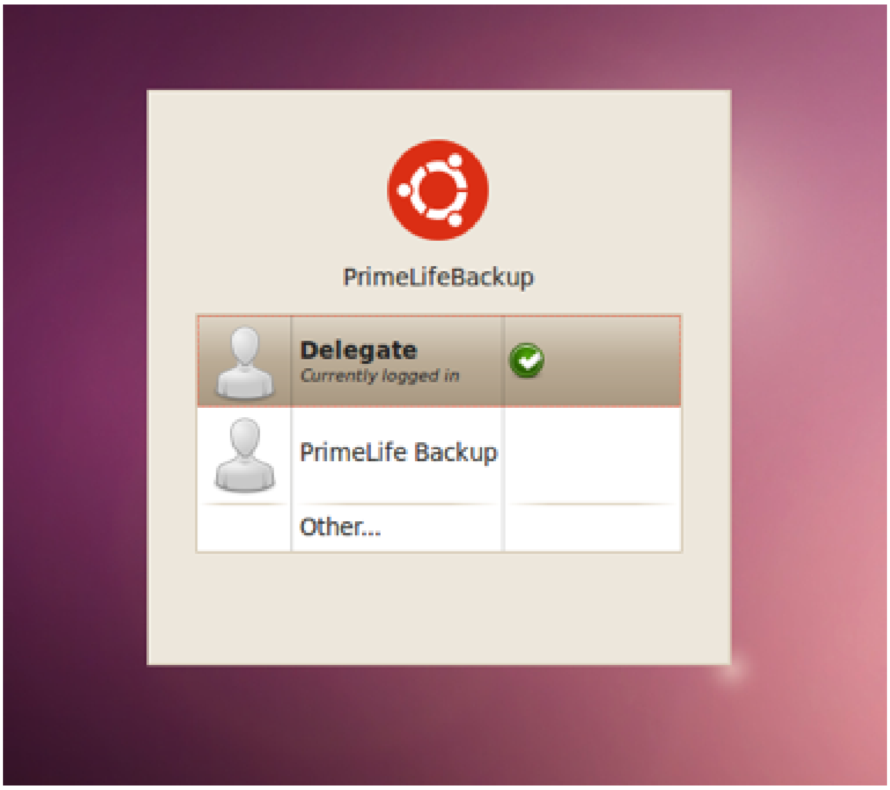


Figure 36: User interface for the delegate.



PrimeLifeBackup

Delegate
Currently logged in ✓

PrimeLife Backup

Other...

Figure 37: Overview about currently logged-in users.

will be more aware of possible actions on her data performed by the delegate.

All in all, the evaluation has shown that the demonstrator is functional and mainly addresses the defined requirements.

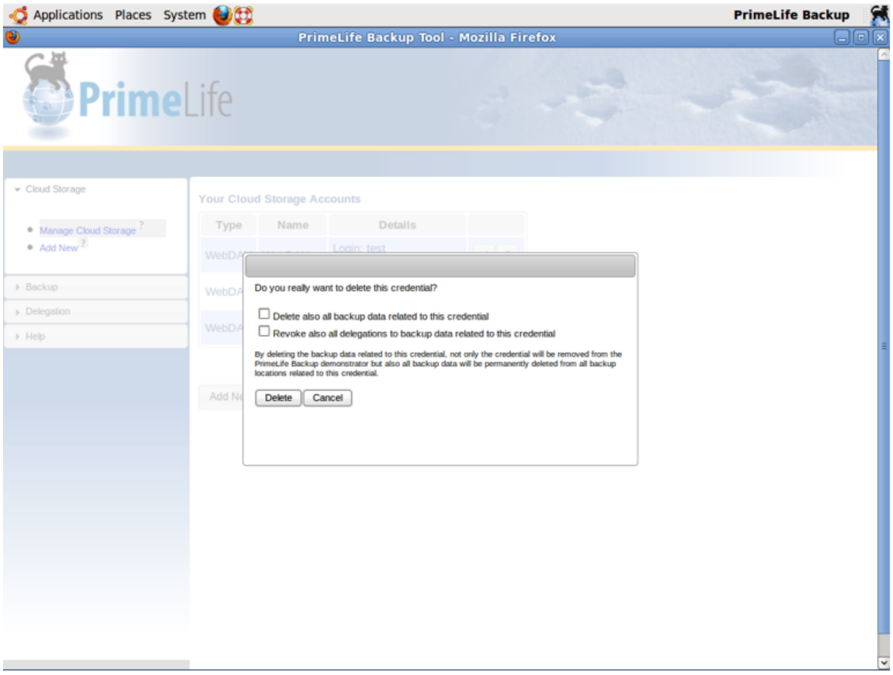
Prototypes have been tested with real users

During the development of the demonstrator, several usability tests have been performed within the PrimeLife project. These tests were performed at different stages of the whole project duration and aimed at improving the demonstrator's usability on the one hand and illustrating possible sources of error on the other hand. Professional feedback by technicians, lawyers and others has been provided, especially by PrimeLife project partners CURE (cf. [WP410]), TILT and ULD. The feedback contained legal evaluations by ULD, technical and usability feedback by CURE and legal feedback as well as technical feedback by TILT. This interdisciplinary expert evaluation from many different fields influenced the further development and the prototype was continuously evolved. Among others, the demonstrator was formatively tested during different project meetings including the latest versions. The feedback gained by this procedure was very precise. Additional feedback was provided from other project partners that haven't been involved in the development and therefore acted like real users (albeit with expert witness). The demonstrator's development was therefore based on longer-term testing. Thus, the requirement of having the prototype tested with real users was sufficiently met in order to improve the interim versions.

Further results of the evaluation

Further requirements that can be defined after the evaluation mainly touch the user interface and not the functionality. For example, the necessity for the identity management system to inform the primary user and supervisory authorities timely on privacy and security breaches and give advice on how to cope with the (potential) consequences can hardly be implemented in a demonstrator. This is because it demands that the storage providers set up appropriate organisational processes that fit into their respective situations - a mere fictitious definition of such a process is clearly out of the scope of the development of a demonstrator. The same applies for the necessity for the identity management system to avoid the use of unique identifiers which may be used in different contexts and to use diverse identifiers where possible. This technical requirement can't be implemented in the demonstrator in a visible and easily checkable manner. Neither is the policy for emergency situation (for example, privacy and security breaches), that is a necessity for a real system, part of the current demonstrator version. This requirement was in detail illustrated in PrimeLife Heartbeat H1.3.7 [WP110c] and further discussed in PrimeLife Heartbeat H1.3.6 [WP110d].

Another requirement that is difficult to implement into the user interface is the necessity for all parties involved to have clarity on the legal, technical and organisational conditions setting the scope for this processing. This would normally be covered by the imprint and the privacy policy of the management system. As the demonstrator does not attempt to be a working system for real users, but instead focuses on offering a basis for the development of a working machine for usage in real life, the demonstrator does



Do you really want to delete this credential?

□ Delete also all backup data related to this credential

□ Revoke also all delegations to backup data related to this credential

By deleting the backup data related to this credential, not only the credential will be removed from the PrimeLife Backup demonstrator but also all backup data will be permanently deleted from all backup locations related to this credential.

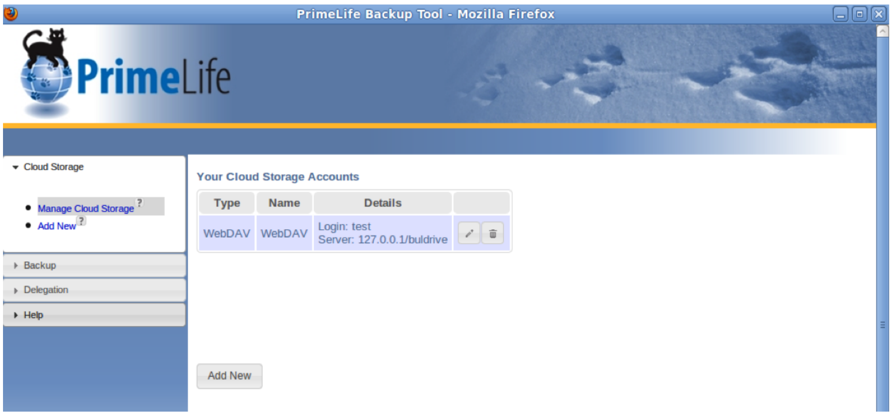
Figure 38: Warning functionality before deleting items.

not contain a privacy policy or an imprint.

The necessity for the identity management system to provide mechanisms for a primary user to get an overview of decisions by her delegate regarding processing of personal data is not implemented as an own item. The same applies for the necessity for the identity management system to provide mechanisms that allow the primary user to trace actions taken by the delegate. This is included in the option of having an all-time overview on issued mandates and on the scope of the mandates. The same applies for the necessity for the identity management system to provide mechanisms that allow the primary user to trace actions taken by the delegate. Anyhow, it would be useful to implement the functionality "delegate's decisions" as an own topic into the user interface. This would allow for quick comprehension of delegate's transactions by the primary user.

The necessity for the identity management system to provide mechanisms for issuance of a mandate, invocation of actions under the name of the primary user with the mandate, verification of the mandate, revocation of the mandate from the delegate and expression of acceptance of the mandate by the delegate (cf. [HRSZ10]) is not sufficiently made explicit in the current implementation. The demonstrator provides information about mandates in general, however, the help functionality should include further information about revoking decisions and consent. Implementing an own button "revoke" would enable the primary user to overrule the delegate's decisions and to revoke consent without intensive searching for the concerning applications.

All the same, the demonstrator already provides necessary information to the primary user before taking important decisions. For instance, the system offers information about potential risks when deleting selected items and reminds the primary user if she really wants to take that decision, cf. Figure 38.



Your Cloud Storage Accounts

Type	Name	Details	
WebDAV	WebDAV	Login: test Server: 127.0.0.1/buldrive	

Figure 39: User interface for cloud storage.

The demonstrator provides the option of using cloud storage for the backup system (cf. Figure 39). In a cloud environment, the data is accessed via information and communications technology using remote hardware instead of being stored only on a local server or computer. The benefits of increased storage at reduced cost allow information to be made readily available [TCl10].

The primary user can select whether she wants to use one ore more storage providers. She can also choose the storage provider for her backuped data from a list, cf. Figure 40.

In PrimeLife Heartbeat H1.3.7 [WP110c] the option of using cloud storage was touched on, but due to the complex legal problems around cloud usage it was suggested to be only a side note at this stage of the development. Anyhow, potential risks of cloud computing have already been analysed and results of such analysis (see inter alia [Wei10]) have been taken into account for the implementation. For instance, the option of storing the backup data in more than one cloud service to avoid total data loss in a worst case scenario is provided in the demonstrator. Besides, the primary user is enabled to choose from a list which storage provider she wants to take. The demonstrator would provide information about the potential risks in case of using a storage provider outside the EU via query functionality as well as information about the storage provider in general without reference to the storage providers' location. The concerning cloud storage provider then offers further information about the terms before contracting with the primary user. Again, for the primary user transparency is granted via the help functionality, cf. Figure 41.

However, this "cloud storage" functionality of the demonstrator does not seem to be different from a general "remote storage" functionality, so it should be checked whether really cloud systems should be addressed here or whether "remote storage" is more fitting term.

Possibilities for improvement

The evaluation of the demonstrator has shown that it implements solutions for the main requirements. Nevertheless, there is still room for improvements. These can be divided

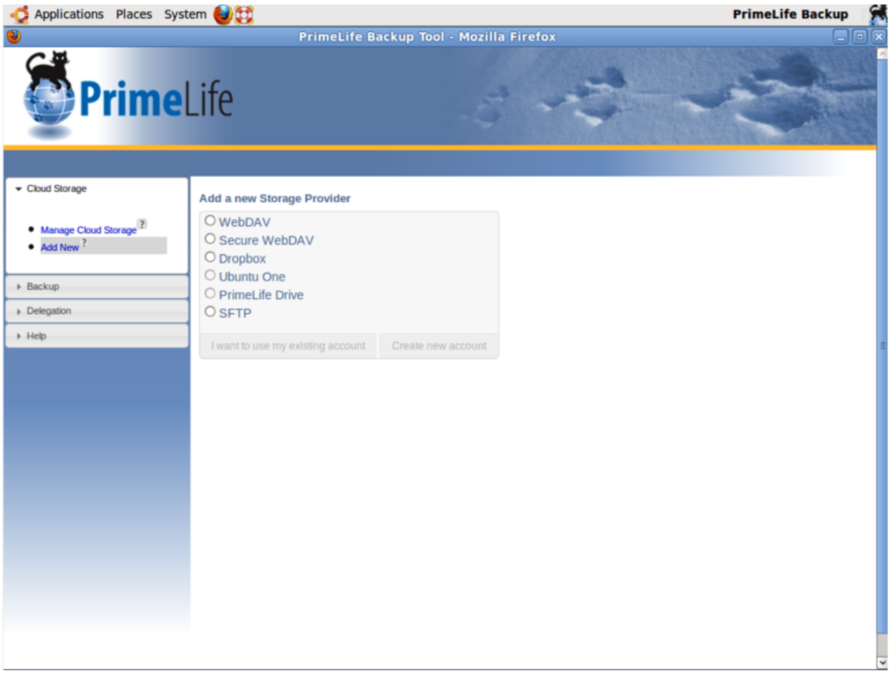


Figure 40: User interface for different storage providers.

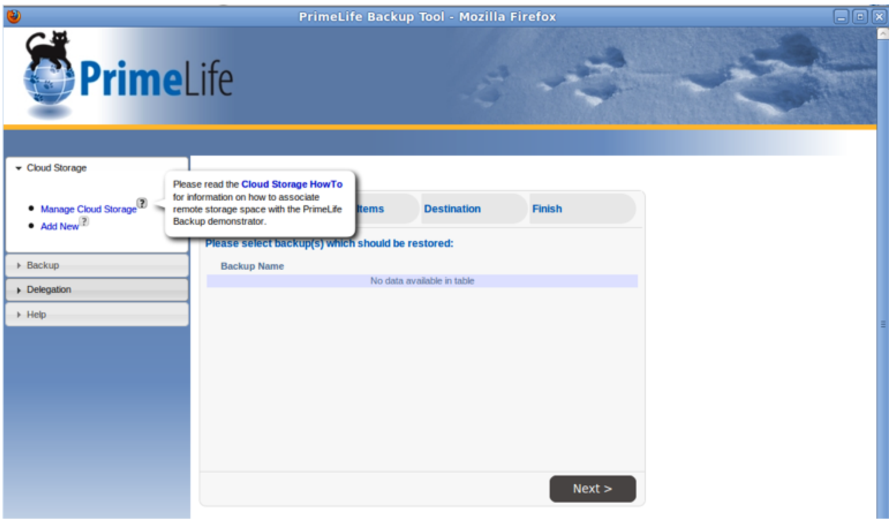
into different areas:

- Requirements that have not been tackled and
- Practical problems.

Requirements that have not been tackled A few requirements that have not been met by the demonstrator in its current status have been discussed in the previous sections. Some of them are not visibly implemented because a visible implementation can hardly be arranged. These are in particular:

- The necessity for the identity management system to inform the primary user and supervisory authorities timely on privacy and security breaches and give advice on how to cope with the (potential) consequences.
- The necessity for the identity management system to avoid the use of unique identifiers which may be used in different contexts and to use diverse identifiers where possible.
- The necessity for the identity management system to have a policy for emergency situation (for example, privacy and security breaches).
- The necessity for all parties involved to have clarity on the legal, technical and organisational conditions setting the scope for this processing.

As illustrated above, solutions for some other requirements can also hardly be visibly integrated into the user interface. This comprises solutions for incident handling, too.



Please read the Cloud Storage HowTo for information on how to associate remote storage space with the PrimeLife Backup demonstrator.

Please select backup(s) which should be restored:

Figure 41: Help functionality within the cloud storage user interface.

As long as there is no incident, the user interface can only provide a blind template. The handling of incidents moreover has to be regulated and predefined in the demonstrator's privacy policy and in internal guidelines. Anyhow, a blind template within the user interface somehow includes the benefit of higher transparency for the primary user: she will be more comfortable and more sensible with a solution that seems to be present within a user interface than with a solution that is only defined within a privacy policy. Thus, neither solutions for the violation of the contract between entities involved are visibly implemented nor solutions for effects of search and seizure. If the contractual relationship between the primary user and the storage provider is terminated, the primary user must maintain the control of her personal data stored at the storage provider. Therefore, regulations for ensuring this requirement have to be covered in the demonstrator's privacy policy and in the demonstrator's terms of use. Also, corresponding solutions are to be established for the cases that the primary user is late with the payment or stops to pay for the service. The same applies for cases of bankruptcy, mergers of corporations, or sales of corporations on the storage provider's side. Also changes of the backup management's privacy policy or terms could lead to an incident that could only be solved within the terms and policy, hardly within the user interface. In case of search and seizure it may be allowed or not allowed to immediately inform other parties about this incident. All parties involved should at least document what is happening when, so that later on (e.g., when a suspect has been cleared) the incident and possible consequences can be reconstructed. Usually the authority in charge has to inform the suspected individuals at least afterwards about the search and seizure procedure, even if they did not notice it and no charges are pressed. This again has to be clarified in the demonstrator's terms.

The same applies for the handling of technical changes. This can neither be visibly implemented in the user interface. It can only be predefined in the specifications of the demonstrator. For example, it is foreseeable that today's assumed strength of cryptographic modules will not be kept of a period of several years. Instead, it will be necessary to migrate to new cryptographic algorithms or other safeguards. Due to the fact that we cannot foresee how these new cryptographic algorithms might look like, solutions for the possibility of migrating the cryptographic functions are only implemented in the back-end but do not have an adequate representation in the user interface. This moreover is a challenge for internal guidelines of the demonstrator.

Additionally, some requirements are not sufficiently implemented because of other reasons. These are in particular: The option for the primary user to revoke her consent or to revoke decisions taken by the delegate is not yet implemented sufficient enough.

Due to the fact, that this was one of the elementary requirements, the demonstrator should be refined in this context. Other requirements that could be better solved are:

- The necessity for the identity management system to provide mechanisms for a primary user to get an overview of decisions by her delegate regarding processing of personal data is not implemented as an own item.

- The necessity for the identity management system to provide mechanisms that allow the primary user to trace actions taken by the delegate is not implemented self-explanatory and easy enough.

- The necessity for the identity management system to provide mechanisms for issuance of the mandate of the delegate, invocation of actions under the name of the principal with the mandate, verification of the mandate, revocation of the mandate from the delegate and expression of acceptance of the mandate by the delegate is not implemented as an own item.

- The necessity for all parties involved to have clarity under which circumstances decisions are revocable/irrevocable and what the potential impact can be. In particular, data controllers should inform primary users on to which degree their decisions are revocable or not.

Solutions for supporting users are at least partially implemented in the demonstrator. Supporting the primary user in sorting her data for example is implemented in the delegation functionality to that extent that the primary user can decide which data sets stored in the backup should be accessible for the delegate. This functionality is part of the backup interface in the “create backup” template and the “manage my backups” template, cf. Figure 42.

Although this interface already provides the option of using partial identities, this could be illustrated more self-explanatory and intuitive. The privacy-enabling social network system Clique developed within the Workpackage 1.2 might serve as an example for effective setting-up of partial identities containing different data. The current status of the demonstrator does not provide a user interface enabling the primary user to select and sort the backuped data that simply and clearly.

Practical problems A general impression of the demonstrator is the fact that the user interface looks already quite professional. The layout is functional and provides much information to the primary user and the delegate. Anyhow, one practical aspect that might cause some problems for a real life usage of the current status is the comparatively high system requirement on the primary user’s machine’s main memory of about 1

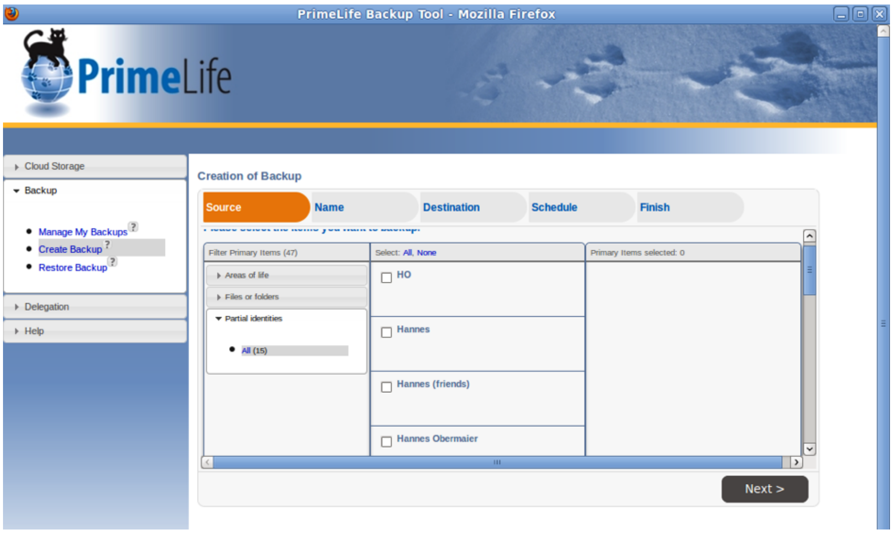


Figure 42: User interface for sorting data.

Gigabyte as well as the file size of more than 3 Gigabyte to download the demonstrator. Especially for older machines this could be too much. If the system requirement is that high, some users interested in using the demonstrator might be discouraged. For real life usage, a further developed demonstrator should therefore demand less extensive system resources.

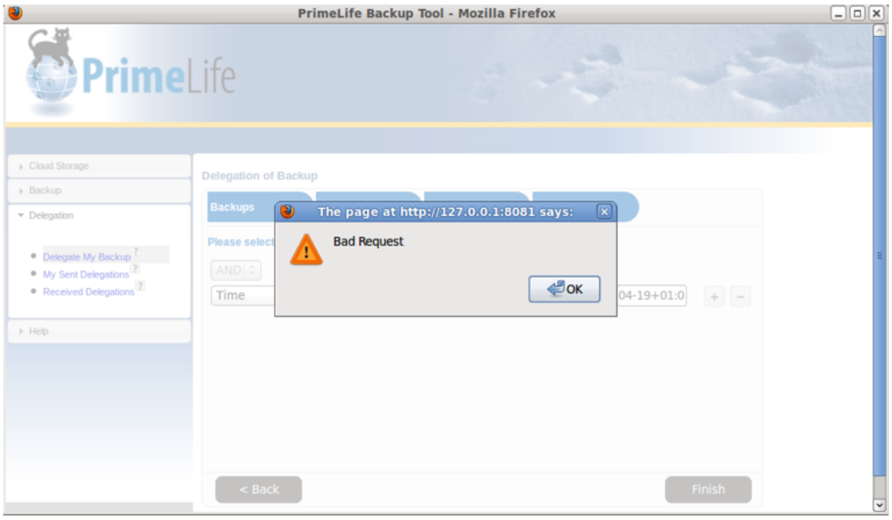
Besides, the current status of the demonstrator sometimes does not seem to be a sufficiently stable system. Warning screens sometimes appear in situations where there should not be ones. For example, warning messages like “Bad request” might occur while trying to finalise the delegation of a backup, cf. Figure 43.

This system instability should be patched; otherwise this would also become an issue for data subjects willing to use the system.

Other aspects of the demonstrator that might need a refinement touches the traceability and the clarity of the stored data sets. It would be useful to provide the information about selected data sets and their content in the file system structure in a tree shape scheme.

Besides, the usability sometimes seems to be improvable. For example, the “create backup” functionality currently requires one click too much: If the primary user starts to create a backup task, she can choose between “areas of life”, “filters or folders” or “partial identities” in the first step “source”. Taking the “areas of life” as an example, the step only offers “all” as a choice. But it needs the next step, clicking on “all” that opens the next window with the fine-granular choice between “family”, “gambling” etc. In this scenario, the differentiation should already be provided in the first step “source” without the necessity of another click. A comparison between Figure 26, Figure 42 and Figure 44 might illustrate this.

An issue of the demonstrator’s graphical layout is the fact that the iFrame does not scale with the screen. This should be refined for a better usability of the demonstrator.



PrimeLife Backup Tool - Mozilla Firefox

The page at http://127.0.0.1:8081 says:

Bad Request

Figure 43: Warning message for failed mandate creation.

Summary

The evaluation of the WP 1.3 demonstrator illustrates that the main requirements for an identity management system developed and defined throughout the PrimeLife project have been implemented. Although the demonstrator did not aim at being a solution for immediate real life usage, the existing functionality enables the demonstrator to be a working system – with little modifications, refining and some specifications.

The demonstrator also covers topics of further importance for identity management systems like the possibility of using cloud storage by providing the option of using such as system.

Due to the fact that the demonstrator is not supposed to be a working system for immediate real life usage but rather the basis for such a system, the evaluation shows that the demonstrator has met the requirements. Taking this and the findings of the demonstrator on the one hand and the additional requirements defined [WP109b], [WP110c] on the other hand as a basis, a working backup and management system can easily be further developed.

The WP 1.3 demonstrator therefore can be the foundation stone for a privacy-enhanced backup synchronization and management system that can be a useful tool for privacy-aware users.

4.3 Concluding Remarks

In this chapter, the concept of the privacy-enhanced backup and synchronisation demonstrator was presented. It was shown that the objectives of lifelong privacy lead to practical results, which can be applied for solving real-life issues in enhanced ways. Our demonstrator reveals new problems that emerge as soon as lifelong aspects related to the data subject are taken into consideration. We presented a new approach, which can help the average citizen to protect herself against unwanted data loss respecting her

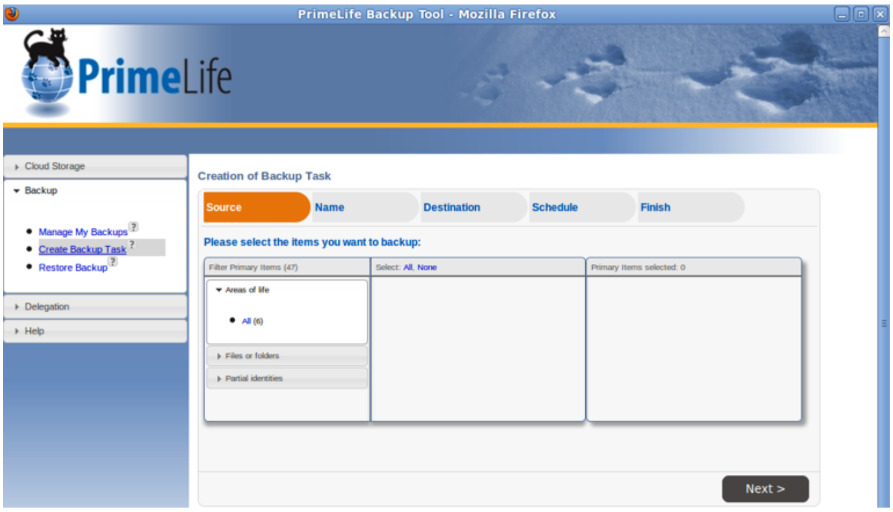


Figure 44: User interface for creating a backup task.

different partial identities and areas of life. Our approach proceeds in such a way that it takes into account lifelong aspects of a human being and the corresponding implications within the scope of privacy.

For many of the envisaged problems that need to be solved with respect to lifelong aspects – and here especially lifelong privacy – solutions are known in principle. However, several concrete implementations that need to be realised in the demonstrator are currently still under development. This, on the one hand, explains why certain aspects and mechanisms were only described at a very high (abstract) level and defines, on the other hand, the research (and development) roadmap for future work in this area.

Chapter 5

Conclusion and Outlook

The mission of Activity 1 of the PrimeLife project is to provide support to retain privacy in the lives of users of Internet technologies. This includes to perform basic research to arrive at a precise understanding of the problems as well as to characterise the space of possible solutions and shortcomings.

In this scope, three particular research areas were addressed within Activity 1 and their corresponding results are presented in this deliverable. The research areas concerned are namely (1) privacy of social software, (2) trustworthiness of online content and (3) identity and privacy issues throughout life. This deliverable presents the final demonstrators and prototypes as well as further relevant results achieved.

In the area of privacy in social software, this deliverable provides an in-depth analysis of privacy issues identified in social software. Furthermore, it comes up with a detailed description of the developed demonstrators related to

- a particular social network site (Clique),
- to the enforcement of selective access control to personal data in social network sites (Scramble!), and
- to selective access control to forum postings (phpBB AC extension)

addressing a selection of the identified privacy-related problems in social software. Moreover, it presents implementation details of the corresponding solutions, their underlying concepts, and the results of user tests performed.

The second research area Activity 1 is concerned with – trustworthiness of online content – contributed to this deliverable with functional mechanisms and requirements enabling to establish trust in the online content. These mechanisms, were implemented as prototypes utilised in investigated scenarios. This deliverable also provides descriptions of the implementation of the prototypes and explains how they are to be utilised.

Last but not least, the third area addressed in Activity 1 – identity and privacy issues throughout life – analysed issues of privacy and identity management throughout life. It elaborates main characteristics and requirements to be fulfilled by according solutions. The demonstrator developed in this scope comes up with the concept of a privacy-enhanced backup solution dealing with aspects of unwanted data loss amplified by the lifelong extent of time. This deliverable presents the conceptual ideas of the proposed solution, its mechanisms as well as the developed focal demonstrator bringing these ideas closer to real end-users.

Concluding, the work in Activity 1 of the PrimeLife project has successfully proven that shifting the control over personal data from administrative parties to the users is (1) a necessary step towards ensuring privacy and (2) feasible with regards to develop suitable technology and training users to take over this control. However, it could also be shown that the paradigm change is not a straight-forward way, but requires thorough consideration of technology designs to make it usable and intuitive. These findings should serve as entry points for further research in this direction.

Also, the discussions about lifelong privacy convey that it represents a much greater challenge than one expects from the first sight. This research area combines a multitude of disciplines including law, technology, and economy and it still has many open research questions to be answered. To give an example, this deliverable touches the concept of delegation legally by providing the current state of the law. However, it also revealed that these laws encounter their limits when delegation is considered triggered by technology. Also, supporting consistent delegation processes and ensuring the privacy of the involved parties challenges more research by computer scientists.

Bibliography

[ACK$^+$09] C.A. Ardagna, J. Camenisch, M. Kohlweiss, R. Leenes, G. Neven, B. Priem, P. Samarati, D. Sommer, and M. Verdicchio. Exploiting Cryptography for Privacy-Enhanced Access Control: A result of the PRIME Project, 2009. to appear.

[Ada99] Anne Adams. The implications of users' privacy perception on communication and information privacy policies. In *In Proceedings of Telecommunications Policy Research Conference*, Washington, DC, 1999.

[AG06] Allesandro Acquisti and Ralph Gross. Imagined communities: Awareness, information sharing, and privacy on the facebook. In *6th Workshop on Privacy Enhancing Technologies*, 2006.

[BBW06] Adam Barth, Dan Boneh, and Brent Waters. Privacy in encrypted content distribution using private broadcast encryption. In *In Financial Cryptography '06*, page 2006. Springer. LNCS, 2006.

[BGW05] D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *Crypto*, pages 258–275, 2005.

[BK09] Jo Bryce and Mathias Klang. Young people, disclosure of personal information and online privacy: Control, choice and consequences. *Inf. Secur. Tech. Rep.*, 14(3):160–166, 2009.

[BMH05] Matthias Bauer, Martin Meints, and Marit Hansen. Structured Overview on Prototypes and Concepts of Identity Management Systems; FIDIS Del. 3.1. Available from http://www.fidis.net/fileadmin/fidis/deliverables/fidis-wp3-del3.1.overview_on_IMS.final.pdf (letzter Abruf 09.02.2009), 2005.

[BMP$^+$09] Patrik Bichsel, Samuel Müller, Franz-Stefan Preiss, Dieter Sommer, and Mario Verdicchio. Security and trust through electronic social network-based interactions. *Computational Science and Engineering, IEEE International Conference on*, 4:1002–1007, 2009.

[Bri98] David Brin. *The Transparent Society: Will Technology Force Us to Choose Between Privacy and Freedom?* Perseus Publishing, 1998.

[Bur09] Jörg Burger. Lügnerin! Betrügerin! http://www.zeit.de/2009/53/Internetmobbing?page=all, December 2009.

[CFHR11] Jan Camenisch, Simone Fischer-Hübner, and Kai Rannenberg, editors. *Privacy and Identity Management for Life.* Springer, 2011.

[Cha85] David Chaum. Security without identification: transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985.

[CHP$^+$09] Sebastian Clauß, Marit Hansen, Andreas Pfitzmann, Maren Raguse, and Sandra Steinbrecher. Tackling the challenge of lifelong privacy. In *eChallenges*, October 2009.

[CK01] Sebastian Clauß and Marit Köhntopp. Identity management and its support of multilateral security. *Computer Networks*, 37(2):205–219, October 2001.

[Con09] G. Conti. Googling security; how much does google know about you? New York, Addison Wesley publishers, p. 91., 2009.

[CvH02] Jan Camenisch and Els van Herreweghen. Design and implementation of the idemix anonymous credential system. In *Proceedings of the 9th ACM conference on Computer and communications security*, pages 21 – 30, 2002.

[Dan09] George Danezis. Inferring privacy policies for social networking services. In *AISec '09: Proceedings of the 2nd ACM workshop on Security and artificial intelligence*, pages 5–10, New York, NY, USA, 2009. ACM.

[Db04] J. Donath and danah boyd. Public displays of connection. *BT Technology Journal*, 22(4):71–83, 2004.

[db08a] danah boyd. Facebook's privacy trainwreck: Exposure, invasion, and social convergence. *Convergence: The International Journal of Research into New Media Technologies*, 14(1):13–20, 2008.

[db08b] danah boyd. *MacArthur Foundation Series on Digital Learning – Youth, Identity, and Digital Media Volume*, chapter Why youth (heart) social network sites: The role of networked publics in teenage social life, pages 119–142. MIT Press, 2008.

[Dir95] 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, 23.11.1995.

[Dör08] Nicola Döring. Reduced social cues / cues filtered out. In N. C. Krämer, S. Schwan, D. Unz, and M. Suckfüll, editors, *Medienpsychologie. Schlüsselbegriffe und Konzepte*, pages 290–297, Stuttgart, 2008. Kohlhammer.

[EGH08] Anja Ebersbach, Markus Glaser, and Richard Heigl. *Social Web*, volume 3065 of *UTB.* UVK, Konstanz, 2008.

[EK02] G. Eysenbach and C. Khler. How do consumers search for and appraise health information on the world wide web? qualitative study using focus groups, usability tests, and in-depth interviews, 2002.

[Eni08] Enisa. Technology-induced Challenges in Privacy and Data Protection in Europe. A report by the ENISA Ad Hoc Working Group on Privacy and Technology, European Network and Information Security Agency (ENISA), Heraklion, Crete, Greece, October 2008.

[FC01] J. W. Fritch and R. L. Cromwell. Evaluating internet resources: identity, affiliation, and cognitive authority in a networked world. *Journal of the American Society for Information Science and Technology*, 52(6):499–507, 2001.

[FM00] A. J. Flanagin and M. J. Metzger. Perceptions of internet information credibility. *Journalism & Mass Communication Quarterly*, 77(3):515–540, 2000.

[FN09] Joshua Fogel and Elham Nehmad. nternet social network communities: Risk taking, trust, and privacy concerns. *Computers in Human Behavior*, 25(1), January 2009.

[Fou07] Eclipse Foundation. Eclipse public license (epl) frequently asked questions, 2007. Accessed Dec. 2007.

[FSD$^+$03] B. J. Fogg, C. Soohoo, D. R. Danielson, L. Marable, J. Stanford, and E.R. Trauber. How do users evaluate the credibility of web sites? a study with over 2.500 participants. proceedings of dux2003, designing for user experiences conference. http://www.consumerwebwatch.org/dynamic/web-credibility-reports-evaluate-abstract.cfm, 2003.

[GA05] Ralph Gross and Alessandro Acquisti. Information revelation and privacy in online social networks. In *WPES '05: Proceedings of the 2005 ACM workshop on Privacy in the electronic society*, pages 71–80, New York, NY, USA, 2005. ACM.

[Gof59] Erving Goffman. *The presentation of self in everyday life*. Doubleday, 1959.

[Gri08] James Grimmelmann. Facebook and the social dynamics of privacy [draft version]. http://works.bepress.com/james_grimmelmann/20/, 2008.

[HBPP05] Marit Hansen, Katrin Borcea-Pfitzmann, and Andreas Pfitzmann. PRIME – Ein europäisches Projekt für nutzerbestimmtes Identitätsmanagement. *it - Information Technology, Oldenbourg*, 6(47):352–359, 2005.

[Hou09] Michelle G. Hough. Keeping it to ourselves: Technology, privacy, and the loss of reserve. *Technology in Society*, 31(4):406–413, 2009.

[How08] Jeff Howe. *Crowdsourcing: Why the power of the crowd is driving the future of business*. Crown Business, 2008.

[HPS08] Marit Hansen, Andreas Pfitzmann, and Sandra Steinbrecher. Identity management throughout one's whole life. *Information Security Technical Report*, 13(2):83–94, 2008.

[HRSZ10] Marit Hansen, Maren Raguse, Katalin Storf, and Harald Zwingelberg. Delegation for privacy management from womb to tomb – a european perspective. In Michele Bezzi, Penny Duquenoy, Simone Fischer-Hübner, Marit Hansen, and Ge Zhang, editors, *Privacy and Identity Management for Life*, volume 320 of *IFIP Advances in Information and Communication Technology*, pages 18–33. Springer Boston, 2010. 10.1007/978-3-642-14282-6_2.

[KPS11] Benjamin Kellermann, Stefanie Pötzsch, and Sandra Steinbrecher. Privacy-respecting reputation for wiki users. In *Proceedings of the 5th IFIP WG 11.11 International Conference on Trust Management (IFIPTM11)*, Copenhagen, Denmark, 2011.

[Lea08] Charles Leadbeater. *We-think: Mass innovation, not mass production.* Profile, 2008.

[Met07] M.J. Metzger. Making sense of credibility on the web: models for evaluating online information and recommendations for future research. *Journal of the American Society for Information Science and Technology*, 58(13):2078–2091, 2007.

[MS09] Viktor Mayer-Schönberger. *Delete: The virtue of forgetting in the digital age.* Princeton University Press, 2009.

[MSF09] Ines Mergel, Charlie Schweik, and Jane Fountain. The transformational effect of web 2.0 technologies on government. http://ssrn.com/abstract=1412796, 2009.

[O'R07] Tim O'Reilly. What is web 2.0: Design patterns and business models for the next generation of software. *Communications & Strategies*, 65(1):17–37, 2007.

[OS08] Kieron O'Hara and Nigel Shadbolt. *The spy in the coffee machine.* Oneworld Publications, 2008.

[PBP10] Stefanie Pötzsch and Katrin Borcea-Pfitzmann. Privacy-respecting access control in collaborative workspaces. In M. Bezzi et al., editor, *Privacy and Identity, IFIP AICT 320*, pages 102–111, Nice, France, 2010. Springer.

[PD03] Leysia Palen and Paul Dourish. Unpacking 'privacy' for a networked world. In *Computer-Human Interaction (CHI) Conference 2003*, pages 129–137, 2003.

[php] phpBB. Official website. http://www.phpbb.com.

[Pöt09] Stefanie Pötzsch. Privacy awareness – a means to solve the privacy paradox? In *Proceedings of the IFIP/FIDIS Internet Security and Privacy Summer School*, 2009.

[PRI] PRIME. Privacy and Identity Management for Europe. https://www.prime-project.eu.

[PWG10] Stefanie Pötzsch, Peter Wolkerstorfer, and Cornelia Graf. Privacy-Awareness Information for Web Forums: Results from an Empirical Study. In *6th Nordic Conference on Human-Computer Interaction*, Reykjavik, October 2010.

[RG10] Kate Raynes-Goldie. Aliases, creeping, and wall cleaning: Understanding privacy in the age of facebook. *First Monday*, 15(1), 2010.

[Ric07] M. Richter, A. Koch. Social software: Status quo und zukunft. *Technischer Bericht*, 2007-01:1–49, 2007.

[Rie02] S.Y. Rieh. Judgement of information quality and cognitive authority in the web. *Journal of the American Society for Information Science and Technology*, 53(2):145–161, 2002.

[SGL06] Martin Szugat, Jan Erik Gewehr, and Cordula Lochmann. *Social Software schnell & kompakt.* entwickler.press, 2006.

[Sol07] Daniel J. Solove. *The future of reputation: Gossip, rumor, and privacy on the Internet.* Yale University Press, 2007.

[Tad10] Monika Taddicken. Measuring Online Privacy Concern and Protection in the (Social) Web: Development of the APCP and APCP-18 Scale. In *60th Annual ICA Conference (International Communication Association)*, Singapur., June 2010.

[Tap09] Don Tapscott. *Grown up digital: How the Net generation is changing your world.* McGraw-Hill, 2009.

[TCl10] Project TClouds. TClouds – Trustworthy Clouds: Privacy and Resilience forInternet-scale Critical Infrastructure, A research project funded by the European Commission's 7th Framework Programme. http://www.tclouds-project.eu/, 2010.

[TS08] Richard H. Thaler and Cass R. Sunstein. *Nudge: Improving decisions about health, wealth, and happiness.* Yale University Press, 2008.

[Tuf08] Zeynep Tufekci. Can you see me now? audience and disclosure regulation in online social network sites. *Bulletin of Science, Technology and Society*, 28(1):20–36, 2008.

[VdBL10] Bibi Van den Berg and Ronald Leenes. Audience segregation in social network sites. In *Proceedings for SocialCom2010/PASSAT2010 (Second IEEE*

International Conference on Social Computing/Second IEEE International Conference on Privacy, Security, Risk and Trust), pages 1111–1117. IEEE Computer Society, 2010.

[VdBL11] Bibi Van den Berg and Ronald Leenes. *Computers, privacy and data protection: An element of choice*, chapter Keeping up appearances: Audience segregation in social network sites. Springer, 2011.

[VdBPL$^+$11] Bibi Van den Berg, Stefanie Pötzsch, Ronald Leenes, Katrin Borcea-Pfitzmann, and Filippe Beato. *Privacy and Identity Management for Life*, chapter Privacy in Social Software. Springer, 2011.

[Wei10] Thilo Weichert. Cloud Computing und Datenschutz, Presentation at the 4th Austrian Day of IT law 2010. English translation "Cloud Computing and Data Privacy" of the presentation script: https://www.datenschutzzentrum.de/cloud-computing/20100617-cloud-computing-and-data-privacy.pdf, 2010.

[WP109a] PrimeLife WP1.2. Requirements and concepts for privacy-enhancing access control in social networks and collaborative workspaces. In Martin Pekárek and Stefanie Pötzsch, editors, *PrimeLife Heartbeart H1.2.5.* PrimeLife, http://www.primelife.eu/results/documents, July 2009.

[WP109b] PrimeLife WP1.3. Requirements and concepts for identity management throughout life. In Katalin Storf, Marit Hansen, and Maren Raguse, editors, *PrimeLife Heartbeat H1.3.5.* PrimeLife, http://www.primelife.eu/results/documents, November 2009.

[WP110a] PrimeLife WP1.2. Privacy enabled communities. In Ronald Leenes Bibi van den Berg, editor, *PrimeLife Deliverable D1.2.1.* PrimeLife, http://www.primelife.eu/results/documents, April 2010.

[WP110b] PrimeLife WP1.2. Privacy-enabled communities demonstrator. In Stefanie Pötzsch, editor, *PrimeLife Deliverable D1.2.2.* PrimeLife, http://www.primelife.eu/results/documents, February 2010.

[WP110c] PrimeLife WP1.3. Second thoughts on the WP 1.3 demonstrator. In Marit Hansen and Leif-Erik Holtz, editors, *PrimeLife Heartbeat H1.3.7.* PrimeLife, http://www.primelife.eu/results/documents, October 2010.

[WP110d] PrimeLife WP1.3. Towards a privacy-enhanced backup and synchronisation demonstrator respecting lifetime aspects. In Jaromír Dobiáš, Katrin Borcea-Pfitzmann, and Stefan Köpsel, editors, *PrimeLife Heartbeat H1.3.6.* PrimeLife, http://www.primelife.eu/results/documents, March 2010.

[WP111] PrimeLife WP1.3. Scenario, Analysis, and Design of Privacy Throughout Life Demonstrator. In Katrin Borcea-Pfitzmann, editor, *PrimeLife*

Deliverable D1.3.1. PrimeLife, http://www.primelife.eu/results/documents, February 2011.

[WP410] PrimeLife WP4.1. High-level prototypes. In Cornelia Graf, Peter Wolkerstorfer, Erik Wästlund, Simone Fischer Hübner, and Benjamin Kellermann, editors, *PrimeLife Deliverable D4.1.4*. PrimeLife, http://www.primelife.eu/results/documents, August 2010.

[WP510] PrimeLife WP5.3. Second Release of the Policy Engine. In Slim Trabelski, editor, *PrimeLife Deliverable D5.3.2*. PrimeLife, http://www.primelife.eu/results/documents, September 2010.

[WP611] PrimeLife WP6.3. Infrastructure for Privacy for Life. In Ulrich Pinsdorf, editor, *PrimeLife Deliverable D6.3.2*. PrimeLife, http://www.primelife.eu/results/documents, January 2011.

[WT99] Alma Whitten and J. D. Tygar. Why johnny can't encrypt: a usability evaluation of pgp 5.0. In *Proceedings of the 8th conference on USENIX Security Symposium - Volume 8*, pages 14–14, Berkeley, CA, USA, 1999. USENIX Association.

[YQH09] Alyson L. Young and Anabel Quan-Haase. Information revelation and internet privacy concerns on social network sites: A case study of facebook. In *C&T 2009*, pages 265–274. ACM, 2009.

[Zim95] Philip R. Zimmermann. *The official PGP user's guide*. MIT Press, Cambridge, MA, USA, 1995.