Privacy and Identity Management in Europe for Life

Open Source Impact

Editors:	Jan Camenisch (IBM)
Reviewers:	Eros Pedrini (UNIMI)
	Harald Zwingelberg (ULD)
Identifier:	D3.3.2
Type:	Deliverable
Version:	0.1
Class:	Public
Date:	May 20, 2011

Abstract

PrimeLife has made available most of its prototypes and demonstrators for download as open source. On top of the PrimeLife has also maintained and evolved code published by the PRIME project. All these open source components have been received surprisingly well. Due to the nature of open source code it is difficult to know for what and how successful it is being used. Still we are aware of surprisingly many new projects that will pick up our results and make use of them. This documents provides an description of the tools and mechanisms made available by PrimeLife and assesses the impact they have had and are expected to have.

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.*

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)**
Open Source Impact	KAU, K.U.Leuven, IBM, TILT, TUD, UNIBG, UNIMI

Executive Summary

PrimeLife set out to develop new solutions for the protection of privacy for the information society and thereby to enable the privacy by design paradigm. An important means to share and promote solutions and concepts is to provide tools and mechanisms in the form of open source. Thus PrimeLife has made available as implementation of as many of its results as possible and reasonable. To this end, a whole work package was dedicated to identify suitable prototype and demonstrators produced by other work packages in the project and then to make them available to the community. Furthermore, the work package had maintained and evolved a selection of the open source projects that were produced by the preceding project PRIME such as the PRIMECore and the identity mixer crypto library. All in all, thirteen tools and mechanisms are now available for download and free use from the PrimeLife project webpages. Of course they vary in terms of code quality from alpha prototypes to code that is mature enough so that commercial products could be build with them.

This documents provides for each of these mechanisms and tools a summary and an assessment of the impact it has already had and what can be expected in the future. Indeed, we find that our results are very well received by the community and that many of them are and will be used in many other projects.

Contents

1 Dudle . . . 10
1.1 Description . . . 10
1.2 Impact . . . 10
2 Over-Encrypt . . . 12
2.1 Description . . . 12
2.2 Impact . . . 13
3 Pri-views . . . 13
3.1 Description . . . 13
3.2 Impact . . . 14
4 Identity Mixer Crypto Library (idemix) . . . 14
4.1 Description . . . 14
4.2 Impact . . . 15
5 Identity Mixer Policy Engine . . . 15
5.1 Description . . . 15
5.2 Impact . . . 17
6 Scramble! . . . 18
6.1 Description . . . 18
6.2 Impact . . . 18
7 Clique . . . 19
7.1 Description . . . 19
7.2 Impact . . . 19
8 Privacy-Oriented Tagging for Clique . . . 20
8.1 Description . . . 20
8.2 Impact . . . 20
9 Privacy Enhancing Browser Extensions . . . 20
9.1 Description . . . 20
9.2 Impact . . . 21
10 PRIMEcore . . . 21
10.1 Description . . . 21
10.2 Impact . . . 22
11 MediaWiki/phpBB – Privacy Enhanced Access Control . . . 23
11.1 Description . . . 23
11.2 Impact . . . 23
12 MediaWiki – Reputation Extension . . . 24
12.1 Description . . . 24
12.2 Impact . . . 24
13 phpBB Privacy-Awareness Support: Personal Data MOD . . . 25
13.1 Description . . . 25

13.2 Impact 26
14 Conclusion 27

Bibliography **29**

List of Figures

1 Screenshot of the Dudle prototype. All votes are stored encrypted on the server. One vote is digitally signed. 11

2 Number of created polls and website accesses of different IP addresses. . . 11

3 Architecture of Identity Mixer Policy Engine. 16

4 Two help pages offered by the PRIMEcore. The overview page (left) and the description of the web service `checkAccess` (right). 22

5 User Interface of MediaWiki History with the Reputation extension 24

6 User interface of the dialog asking for the user's reputation value 25

7 User interface of Personal Data MOD. 26

8 Eye icons in different shades of red representing four visibility classes for personal data. 27

Open Source Impact

1 Dudle

1.1 Description

This Web 2.0 application offers functionality to make small polls. These polls either may have the target to schedule some event (e.g., schedule a TelCo, meeting etc.) or asks for some alternative things (e.g.,find out which sort of coffee some people like the most).

Unlike most other similar applications, Dudle offers enhanced access control settings, which minimizes trust assumptions into the polling server. Users can configure their poll, depending on whom they trust. Trust scenarios, which are covered by the prototype are:

- trust in all participants of the group,
- trust in the poll initiator only,
- minimal trust in all parties.

A screenshot of the vote interface is shown in Figure 1. There, the small locks beneath the user names indicate, that the single votes are symmetrically encrypted. Due to the use of JavaScript, the user does not have to install additional software (zero footprint application).

1.2 Impact

Two different kind of users may profit from this prototype. First, end users who want to use an event-scheduling application can create their specific poll at the server we set up within PrimeLife. Second, server administrators may set up their own server.

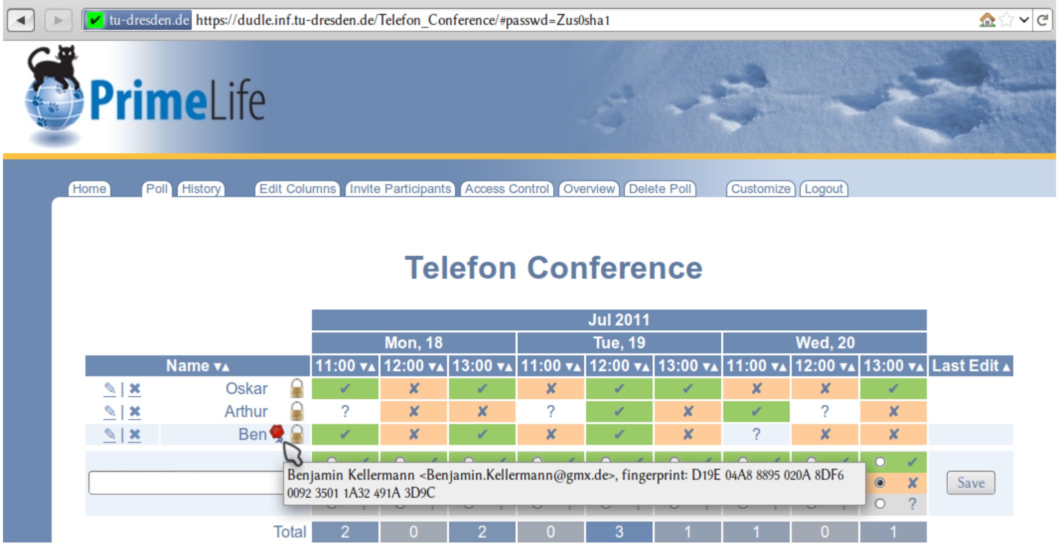
To enhance the acceptance of the prototype, PrimeLife used the fact, that several languages are natively spoken within the project and with the help of all partners, the application was translated into 7 languages. The Dudle server set-up from PrimeLife currently has about 2000 requests from different IP-addresses per month, about 80 new polls are created per month.[1]

Figure 2 shows a statistic of how many polls where created between December 2009 and April 2011.[2] In addition to poll-creation, the red plot shows, how many different IP addresses access the Dudle server.

To improve the value of the application for the OpenSource community and therefore other server administrators, several mechanisms were implemented:

[1] This number does only include real polls and not the significant higher number of test polls.

[2] 67 % of the polls were deleted from the statistics which we claim to be dummy and test polls. These include polls with less than 3 participants, less than 3 different time slots, and polls which had less than 10 changes.



Telefon Conference

	Jul 2011									
	Mon, 18			Tue, 19			Wed, 20			
Name ▾▴	11:00 ▾▴	12:00 ▾▴	13:00 ▾▴	11:00 ▾▴	12:00 ▾▴	13:00 ▾▴	11:00 ▾▴	12:00 ▾▴	13:00 ▾▴	Last Edit ▴
✎ | ✖ Oskar	✓	✗	✓	✗	✓	✓	✗	✗	✓	
✎ | ✖ Arthur	?	✗	✗	?	✓	✗	✓	?	✗	
✎ | ✖ Ben	✓	✗	✓	✗	✓	✗	?	✗	✗	
Total	2	0	2	0	3	1	1	0	1	

Benjamin Kellermann <Benjamin.Kellermann@gmx.de>, fingerprint: D19E 04A8 8895 020A 8DF6 0092 3501 1A32 491A 3D9C

Figure 1: Screenshot of the Dudle prototype. All votes are stored encrypted on the server. One vote is digitally signed.

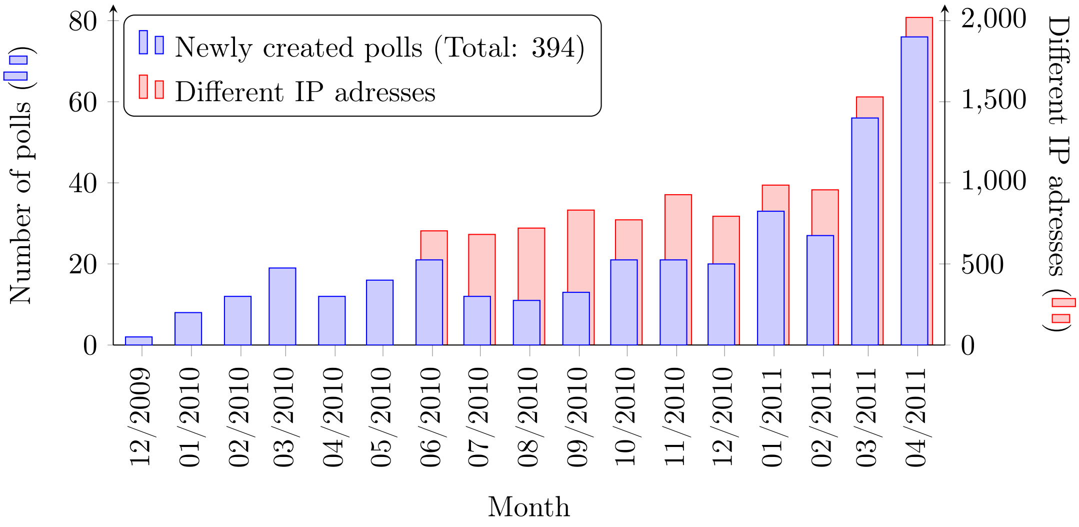


Figure 2: Number of created polls and website accesses of different IP addresses.

- The application is extendable through an extension mechanism. This enables the community members to add functionality without the need to dive too deep into source code details of the application.

- The look and feel of the service can be adjusted by adding according CSS files. Dudle searches through a dedicated directory and uses the styles found there.

- The well-known GetText localisation mechanism was integrated. As already stated, the PrimeLife partners created language files for several languages. The French translation was submitted by two translators from the OpenSource community.

The source code of the application can be downloaded in form of "latest snapshots" or branched using a version control system. All source code is available under the AGPLv3 license.

To increase the publicity of the prototype, it was posted at the well-known OpenSource sites freshmeat[3] and alternativeTo[4]. As of May 2011, there are at least 6 other Dudle servers, which were set-up by members of the OpenSource community, which shows its acceptance.

2 Over-Encrypt

2.1 Description

This tool provides a secure solution for data sharing capabilities in an outsourcing scenario where the storage service provider is trusted neither for data confidentiality nor for enforcing access control functionalities.

The tool implements the approach for the encryption of outsourced data and key management developed within WP2.4 by UNIBG and UNIMI [DFJ$^+$10b, DFJ$^+$10a]. The tool consists of a client and a server component. The approach allows regulating access to external resources stored by honest-but-curious servers by translating the authorization policy into an equivalent encryption policy: stored resources are encrypted in such a way that only users authorized to read the resources will be able to decrypt them. To ensure each resource to have only one encrypted occurrence (regardless of the number of users authorized for it), and each user to have only one key (regardless of the number of resources the user is authorized to access), the approach uses a hierarchical organization of the keys together with token-based key derivation. The approach exploits on a double layer of encryption (one controlled by the client and the other by the server) to enable enforcing policy updates directly at the server (by properly changing the encryption server-side) without need for the client to download, re-encrypt, and re-upload resources to reflect authorization changes. The first version of the tool used as client a Firefox plugin that permits users to exchange resources that are kept encrypted on the server. The enhanced version of the tool is integrated within the Nautilus file manager. With this client, the services of the outsourcing server can be invoked using an approach similar to classical file management. Users can easily upload and download files using the known tools of the file manager. The protocol for the communication with the server has

[3] http://freshmeat.net/projects/dudle

[4] http://alternativeto.net/software/dudle/

been extended and relies on WebDAV. Apart from the extension of the communication protocol over WebDAV, there was no need to apply modifications to the server.

2.2 Impact

The use of providers for the storage and dissemination of resources over the Internet has seen a significant increase in the past few years (e.g., Dropbox, Google Docs, Microsoft Office Live). These applications are considered by many a significant threat to the privacy of users, due to the lack of confidentiality guarantees. Competitors to existing solutions are emerging that distinguish themselves for the encryption protection that relies on keys stored only on the client (e.g., SpiderOak). These solutions do not typically offer functionality for the protected sharing of resources among users, and no solution currently offers mechanisms that avoid the re-upload of a resource when the access privileges change with the same guarantees offered by over-encryption. The open source Over-Encrypt tool has the potential to support developers who want to further extend the functionality of current systems for the protected online sharing of resources. The availability of two implementations of clients for the server, both for a browser and for a file manager, helps to demonstrate the flexibility of the approach and its adaptability to several scenarios. The two clients also represent the most natural scenarios where the services will be used.

To increase the visibility of this prototype, it has been uploaded to the well know open source repository SourceForge (`http://http://sourceforge.net/projects/over-encrypt/`).

3 Pri-views

3.1 Description

Pri-views is a tool that on input

- a relational table,
- a set of confidentiality constraints, specifying sensitive attributes or sensitive attribute associations, and
- a set of visibility constraints specifying requirements for attributes or attribute associations to be made visible,

computes data views (vertical fragments of the table) that ensure no confidential information is released and required visibility over the data is satisfied.

Pri-views is based on a so-called greedy algorithm designed by UNIBG and UNIMI to solve the problem of creating unlinkable fragments in the storage of sensitive attributes [CDF$^+$09]. The approach builds on previous work on protecting sensitive information in external data storage but, unlike previous proposals, completely departs from encryption simply relying on fragmentation as a means for protecting confidentiality. The approach is then applicable to data publishing scenarios where the data holder may be requested to produce views over the data for external observers while ensuring confidential information is properly protected. The tool is composed of two applications: the first implements the proposed greedy algorithm (developed in C++), while the second realizes its graphical user interface (developed in Java).

3.2 Impact

The tool demonstrates how the fragmentation techniques can be applied over large schemas of relational tables. The users that can be interested in the tool are database and security experts who want to verify the behavior of the fragmentation techniques developed within PrimeLife. The model used by the tool is abstract and its services can be immediately adapted for an integration within any software environment supporting the design of relational schemas. Implementors of these systems can consider the reuse of the tool within those design environments.

To increase the visibility of this prototype, it has been uploaded to the well know open source repository Google code (`http://code.google.com/p/pri-views/`).

4 Identity Mixer Crypto Library (idemix)

4.1 Description

We all increasingly use electronic services in our daily lives. To do so, we have no choice but to provide plenty of personal information for authorisation, billing purposes, or as part of the terms and conditions of service providers. Dispersing all these personal information erodes our privacy and puts us at risk of abuse of this information by criminals. Identity Mixer (idemix) allows users to minimise the personal data they have to reveal in such transactions. For instance, if electronic identity (eID) cards were realised with idemix, then teenagers possessing such eID cards could log onto a teenage chat room just proving that they are indeed 12-15 years of age without revealing any other information stored on the card such as their name or address.

The Identity Mixer cryptographic library offers the all the cryptographic algorithms to realise such anonymous authentication. This comprises the functionality for the issuer, client, and service provider. The library implements the credential system of Camenisch and Lysyanskaya [see idemix.wordpress.com for further information]. In addition to the basic credential system, the following additional features are currently supported for dealing with attributes contained in a credential when proving possession of credentials:

- Selective release of the attributes (minimal disclosure);
- Proving predicates over some of attributes; and
- Verifiable encryption and anonymity revocation (useful for conditional anonymity) of some attributes

On top of that, the library also allows for proofs of possession of several credentials at the same time and to state various relations among the attributes contained in these credentials. For instance, as user could proof that she holds a driver's license as well as a student ID card and that she has made the driver's licence more that four years ago.

The library provides algorithms to generate all key material, to issue credentials, and to demonstrate possession of credentials. To realise a typical application, these algorithms need to be embedded into an access control system, similarly as the algorithms to generate and verify, e.g., x.509 or SAML token would need to be embedded. We refer to the documentation of the library for further information. As one example use case, idemix has been integrated into the PRIME Core privacy-enhancing identity management system.

4.2 Impact

The Identity Mixer crypto library has been publicly available with source code under a license that allows for free commercial use for a while now. Before that, it has already been made available to interested parties on a individual basis since about 2004.

We know that a number of universities worldwide have built demonstrator and pilots with idemix. For instance, KU Leuven's Distributed Systems research group made a couple of pilots (see, e.g., [VVL^{+}10]), University of Groningen has done a smart-card integration [Dan07].

Finally, identity mixer will be used in a number of new research projects including the following EU funded ones ABC4Trust (realizing to pilots – `www.abc4trust.eu`), di.me (identity mangement – `dime-project.eu`), and FI-Ware (which builds the future internet infrastructure – `fi-ware.morfeo-project.org`).

5 Identity Mixer Policy Engine

5.1 Description

This software package extends the Identity Mixer library (which was already made available open-source under the PRIME project) with an intuitive and flexible policy language. Using this language, relying parties can express which credentials a user has to present, which attributes of those credentials have to be revealed, and which conditions the attributes have to satisfy. The package also defines a corresponding claims language, by means of which the user communicates to the relying party the actual identity properties being proved by the Identity Mixer token. The policy language makes it much easier for relying parties to support Identity Mixer credentials as a valid means of authentication and write their own access policies matching their specific needs. The language is based on the card-based access requirements language (CARL) developed in Activity 5 [CMN^{+}10], supporting advanced features such as selective revealing of attributes, policies and claims involving multiple cards, and proving conditions involving attributes from different cards.

The policy engine implements most of the steps in the typical authentication interaction depicted in Figure 1. Initially, the user contacts a server to request access to a resource she is interested in (1). Having received the request, the server responds with the applicable access control policy for the resource (2). The applicable policy contains the card requirements expressing which cards have to be presented, which attributes on those cards have to be revealed, which conditions the attributes on the cards have to satisfy, and which entities are considered trusted issuers for these cards. Upon receiving the policy, the user's system searches the user's card store for combinations of cards that fulfill the policy (2a) and for each combination, derives the claim that will be revealed if this combination is chosen. Once the user indicates (2b) which combination of cards is to be used for the authentication, the cryptographic evidence for the chosen claim is generated (2c). The claim and the evidence are sent to the server (3). Finally, the

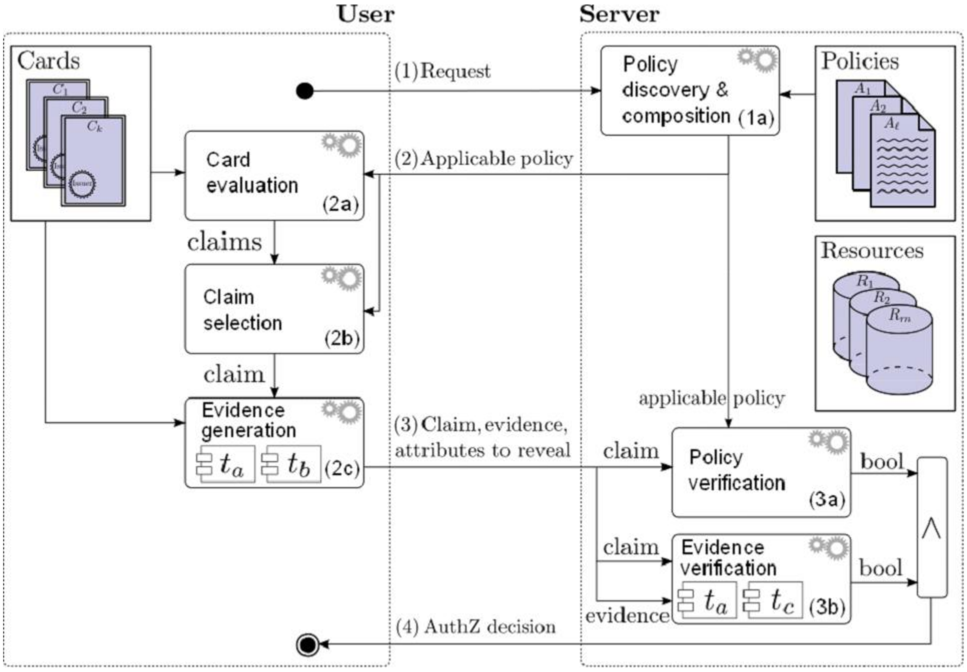


Figure 3: Architecture of Identity Mixer Policy Engine.

server verifies whether the claim satisfies the policy (3a) and whether the cryptographic evidence supports the claim (3b). If so, access to the resource is granted (4).

The current software package implements steps (2a), (2c), (3a), (3b) in the above scenario. Meaning, given an access control policy expressed in CARL (2), the package parses a user's credential store and returns the list of card combinations and associated claims that can be used to satisfy the policy (2b). For a particular chosen claim, the engine invokes the Identity Mixer library and generates an Identity Mixer token supporting the chosen claim (2c). It also implements the server's functionality of checking whether the transmitted claim fulfills the required policy (3a) and verifying that the Identity Mixer token supports the claim (3b). The policy engine does not, however, provide a user interface to facilitate the claim selection process in step (3b), and neither does it implement the policy discovery step (1a). For the former, an existing identity selection based on the open-source project Higgins or on Microsoft's Cardspace would have to be integrated with the engine. For the latter, one could use any access control engine, e.g., based on XACML.

The CARL language is very intuitive and does not require a deep understanding of the underlying crypto, thereby considerably lowering the threshold for relying parties to write their own policies and adopt Identity Mixer technology. As an example, we provide a sample CARL policy for a car rental scenario below. The policy requires the user to have a membership card, a credit card, a driver's license, and an insurance statement. The user is required to reveal her credit card number and insurance policy number, and has to convince the server that

- she has a license for driving vehicles of category B
- she has insurance covering for amounts over 30000 USD
- she has gold or silver membership status
- her credit card has not expired
- her name on the membership card matches that on her driver's license.

The corresponding policy in CARL is:

```
own mc of-type MemberShipCard issued-by CarRentalCo.com
own cc of-type CreditCard issued-by Amex.com, Visa.com
own dl of-type DriversLicense issued-by DeptMotorVehicles.com
own is of-type InsuranceStmt issued-by InsuranceCo.com
reveal cc.number
reveal is.policyNo
where dl.vehicleCategoryB == true and
      is.guaranteedUSDAmout >= 30000 and
      (mc.status == "gold" or mc.status == "silver") and
      cc.expDate > today() and
      mc.name = dl.name
```

In response, the user may send the following claim based on the cards in her card store, with a matching Identity Mixer token as evidence:

```
cc.number -> 1234 5678 9012 3456
is.policyNo -> 12345
own mc of-type MemberShipCard issued-by CarRentalCo.com
own cc of-type CreditCard issued-by Amex.com
own dl of-type DriversLicense issued-by DeptMotorVehicles.com
own is of-type InsuranceStmt issued-by InsuranceCo.com
where dl.vehicleCategoryB == true and
is.guaranteedUSDAmout >= 30000 and
mc.status == "silver" and
cc.expDate > today() and
mc.name = dl.name
```

5.2 Impact

As the Identity Mixer Policy Engine was only made available by the very end of the PrimeLife project, as of yet we do not have any representative statistics on the number of downloads, reactions, or adoption plans within the community. We do believe, however, that the availability of an intuitive policy language that abstracts away from all the cryptographic details will considerably lower the threshold for non-specialists to adopt Identity Mixer technology to enforce access control requirements that match their specific needs.

Moreover, some concepts of the CARL and PPL languages are already finding their way for standardization within OASIS. In particular, we have proposed a profile to SAML that allows identity providers to certify that a user's attributes satisfy a given predicate, instead of always having to reveal the full values of these attributes. Our profile was well received by both the SAML and XACML technical committees, so we are optimistic about being able to reach standardization soon. On the longer term, the technical committees also expressed interest in some of the more ambitious concepts of the CARL and PPL languages, so we will continue to strive to let our ideas influence future versions of SAML and XACML.

Finally, at least part of this implementation will be used in the EU-funded projects that make use of the Identity Mixer crypto library (cf. previous section).

6 Scramble!

6.1 Description

"Scramble your social network data!" – With Scramble you can selective enforce you access control preferences for your content on social networks such as Facebook or Twitter.

Scramble! provides a mechanisms for users to enforce access control over their own data. Its main target is to protect users from sharing sensitive information with Social Network Sites (SNS) providers. Scramble! allows the creation of a web of friends and by means of encryption to enforce access control. Scramble! makes it possible to create different groups of friends with separate access to sensitive data and allows for on-the-fly decryption in the background.

Scramble! uses the OpenPGP standard as the encryption mechanism, and therefore builds further on the existing PKI infrastructure and key management model, allowing also "broadcast" encryption for multiple recipients. GnuPG also allows to perform anonymous recipient encryption by omitting the public key IDs from the encrypted blob, but is still vulnerable for active attacks. Apart from storing large amounts of encrypted data in a SNS, it is also possible to only list "tiny url" snippets, that refer to the encrypted data into a third server. In this way the problem of the large ciphertext size that currently grows linear in the number of users that are granted access is minimised, as well as the visual contamination of the SNS platform.

6.2 Impact

We have made the Scramble! application available to the OpenSource community at the very end of the PrimeLife project as a SourceForge project under the EPL license. Dissemination work was also done with local students and during some conferences, presenting live demos with good general reactions as a result. The application so far didn't have representative contributions from the OpenSource community due to the fact that the source code is not yet available, which will be done at the end of the project. However, some downloads have been made already from users unrelated to PrimeLife. We believe that Scramble! will find good adoption after its full release to the OpenSource community and will force social network providers to react to it as it will directly affect their revenues.

7 Clique

7.1 Description

In many current-day social network sites privacy problems are a real issue. They have a number of origins and causes, and can take a variety of forms. One of the commonest problems is the fact that users of social networks share too much information, or share information with the wrong audience. Often, they share information with a certain audience in mind, while in practice a much larger audience has access to it. Moreover, users often threaten other users' privacy by disclosing information about them to unintended audiences. In the PrimeLife project, we concluded that one of the central social mechanisms that individuals use in the real world to present good impressions to different audiences is absent in the online world. Following the twentieth century sociologist Erving Goffman, we concluded that individuals use "audience segregation" in the 'offline world' to ensure that they can disclose different information about themselves to different audiences, and to guarantee that information from one social setting, which may negatively impact their image in another social setting, does not seep into the latter. For example, individuals show different sides of themselves when in a professional context than when at home with their family. Mechanisms for audience segregation are lacking in the online world, and this may cause privacy problems, particularly in social network sites, where so much (personal) information is shared. This tool was created to remedy this gap. To realise audience segregation in a social network site, researchers at TILT built a platform called Clique, in which they implemented the following mechanisms:

- collections: users can cluster contacts into meaningful lists, to which they can give access rights for viewing information or content
- faces: users can create different 'faces' within the same social network site, for example presenting separate professional and personal profile pages, which contain different contacts, and hence present different sides of themselves.

Clique is a modification of the Elgg social networking platform. Clique provides users with a social network platform that enables them to keep control over their privacy. This includes, for example, fine grained access control and configuration of multiple faces (e.g. family, personal, professional) that can be used for interactions with other users. When posting a data item, e.g., name, birthday or profile photo on the site, the user can define for every single other user whether they should be able to see it or not.

7.2 Impact

Clique has had impact on many different levels. While it started out as a demonstrator, almost a proof of principle, to show users and providers that (more) privacy-friendly ways of enabling social networking are feasible, in the 14 months since the tool has gone 'live' almost 2000 users have created an account to see this alternative social network site. The impact in terms of user interest, therefore, has been much greater than we had originally expected. Second, Clique has had considerable impact in terms of media interest. After its launch in the spring of 2010 there was significant media coverage, especially in the Netherlands, where Clique was developed. National and regional newspapers reported on this new, privacy-enhanced social network site, and the principle researchers gave several radio interviews as well. This led to exposure to a larger audience, and resulted, among other things, in a meeting with one of the largest internet providers in the Netherlands, and in an invitation to discuss privacy issues in social network sites (and the tool itself) at a high school with several hundred teenagers. Third, throughout the last year the researchers added extra tools to the system to further increase privacy-friendliness and usability. These include two wizards to create collections and faces, and a movie explaining the system's underlying idea(l)s and workings. A recent quantitative study of the system's log files and an implemented statistics tool reveal that these additions have had considerable impact: the movie has been viewed 1668 times to date, while the wizards have been accessed 2178 times.

8 Privacy-Oriented Tagging for Clique

8.1 Description

One of the most important features of many web 2.0 environments is tagging. Tagging a resource helps to describe the resource itself and allows it to be retrieved through browsing or searching. Tags are generally chosen informally and personally by the resource owner or by its viewers, depending on the system. While convenient since they are an easy and efficient way of adding metadata to content, tagging functions may cause privacy problems. One area where these problems are particularly relevant is in social applications (e.g., social networks), especially when individuals are tagged on a resource that doesn't belong to them. The Privacy-Oriented Tagging System developed by UNIMI, allows users tagged on a resource in social networks participate in the control of the resource. The tool is a plug-in of Clique that enforces an access control layer where the users tagged within a resource are the main actors in defining the access control policies for that resource.

8.2 Impact

The tool demonstrates how to implement an access control system designed to increase the users privacy protection, supporting users notification and specific access control policies for tagging. The model used by the tool is abstract and its services can be immediately adapted for an integration within tagging system of other social network sites. In fact this tool has been designed to be compatible with both Clique and Elgg social networks, and this can represent an important point of contact with the big open source community of Elgg.

9 Privacy Enhancing Browser Extensions

9.1 Description

The Privacy Dashboard is a Firefox add-on designed to help you understand what personal information is being collected by websites, and to provide you with a means to control this on a per website basis.

The Dashboard installs an icon on the browser's navigation toolbar that changes according to the current website:

- websites that take good care of your privacy
- websites which collect some information, but lack a machine readable privacy policy
- websites that enable third parties to track you across the Web

The Dashboard alerts you the first time you visit a website (except for nicely behaved ones) giving you the opportunity to set your preferences. Thereafter you review these by clicking the Dashboard icon (or the Tools|Privacy Dashboard menu item) when visiting the website.

The 'Data Track' tab allows you to make a range of queries about sites. The 'Location' tab deals with your physical (geo)location. The 'Current Website' tab provides information on the current site, and allows you to review and update your preferences. Note that the Dashboard's assessment of a website provides no guarantees as to the privacy friendliness of that site. You are advised to check the site's privacy policy.

9.2 Impact

The Dashboard is now an open source project hosted by W3C and we are seeking volunteers to help with adding new features, bug fixes, localizations and community-led documentation. The project site is at:

- `http://code.w3.org/privacy-dashboard`

The Dashboard was extended to automatically survey the home pages for the top one thousand sites (according to Google) and the data gathered can be queried online at the following URL:

- `http://www.w3.org/services/privacy-dashboard/`

In addition, the Privacy Dashboard allows its users to opt in to sharing the data they collect as they browse the Web. The anonymized data will be periodically uploaded to a user specifiable site, which defaults to the W3C. This is expected overtime to provide a wealth of data on the evolving privacy practices of websites, and to feed into further work on improving the way privacy is managed.

10 PRIMEcore

10.1 Description

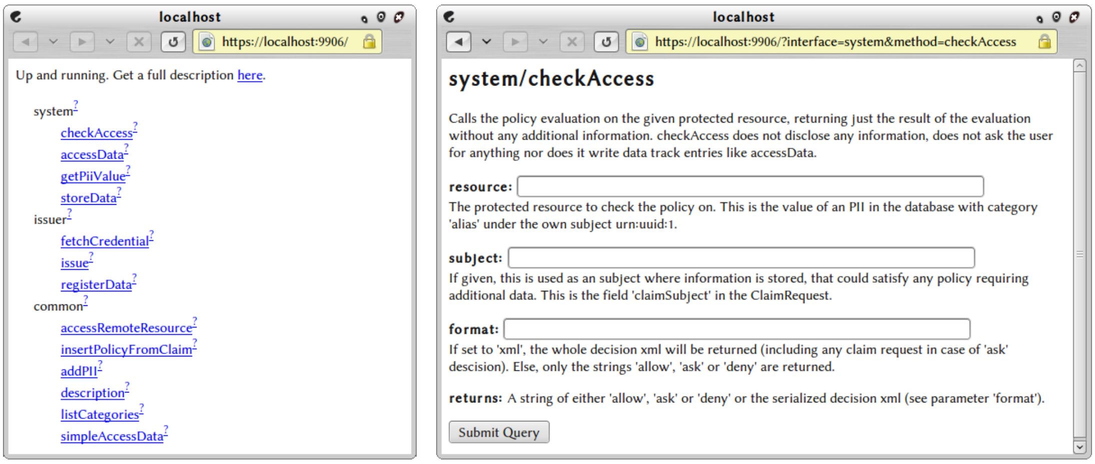
Web applications dealing with personal data in a privacy-friendly way have the need for anonymous credential systems. While there are already protocols describing anonymous credential systems and libraries, implementing the protocols, application using the libraries are still rare. PRIMEcore offers a set of useful web services. They exchange information by using cryptographic protocols. It contains all necessary functionalities to run it on both client as well as server side.

Thus without applications supporting anonymous credentials, companies will not start building a credential infrastructure and vice versa. Using the Identity Mixer Crypto Library, the PRIMEcore implements an easy way to issue and use anonymous credentials for web applications. By reducing the initial cost for both parties, the barrier of “starting first” can be lowered. In addition, the user side of the PRIMEcore keeps track of data, which has been released to third parties.

Therefore, the intended audience of the PRIMEcore includes

- Web-developers, who want to use anonymous credentials in an easy way,
- Java-Developers, who want to have an example implementation for the Idemix library, and
- Java-Developers, who want to use this as a starting point for a full featured IdM solution.

To make development easy, a tutorial has been written,[5] which addresses the main points in developing applications with the PRIMEcore. In addition, the on-line help system (cf. Figure 4) of the PRIMEcore can be used while developing applications.



Up and running. Get a full description here.

system?
checkAccess?
accessData?
getPiiValue?
storeData?
issuer?
fetchCredential?
issue?
registerData?
common?
accessRemoteResource?
insertPolicyFromClaim?
addPII?
description?
listCategories?
simpleAccessData?

system/checkAccess

Calls the policy evaluation on the given protected resource, returning just the result of the evaluation without any additional information. checkAccess does not disclose any information, does not ask the user for anything nor does it write data track entries like accessData.

resource:
The protected resource to check the policy on. This is the value of an PII in the database with category 'alias' under the own subject urn:uuid:1.

subject:
If given, this is used as an subject where information is stored, that could satisfy any policy requiring additional data. This is the field 'claimSubject' in the ClaimRequest.

format:
If set to 'xml', the whole decision xml will be returned (including any claim request in case of 'ask' descision). Else, only the strings 'allow', 'ask' or 'deny' are returned.

returns: A string of either 'allow', 'ask' or 'deny' or the serialized decision xml (see parameter 'format').

Figure 4: Two help pages offered by the PRIMEcore. The overview page (left) and the description of the web service `checkAccess` (right).

To enhance the value of the PRIMEcore, it can be extended in two ways. First, the Send Personal Data Dialog may be changed by developers easily, without the need to change the source code of the application. Additionally, the underlying ontology can be dynamically changed and therefore adopted to other scenarios.

10.2 Impact

The PRIMEcore is foremost a result of the PRIME project but was maintained and extended by PrimeLife. It has mainly been used within the PrimeLife project as a basis

[5] `http://prime.inf.tu-dresden.de/core/doc`

of the prototypes within Activity 1. We hope that other projects will find it also usable for their purposes, either as a whole or some of its components. Of particular interest thereby might be the Send Personal Data Dialog and the Data Track, the latter seems to find its way into other projects, but unfortunately nothing concrete can be said at the time of this writing.

11 MediaWiki/phpBB – Privacy Enhanced Access Control

11.1 Description

MediaWiki is a free Wiki software, which is used for many open source wikis (e.g., Wikipedia), whereas phpBB is a popular software for internet forums.

Both, wikis and forums, enable and encourage users to create and share content with a broad community. Thereby, content creators usually have no control who has access to their contribution, i.e. their wiki articles or forum posts. In PrimeLife, we developed two extensions – one for MediaWiki and another one for phpBB – that upgrade the access control features of MediaWiki/phpBB forum software so that users, instead of administrators, can define who should have access to their own contributions.

Since in a wiki or forum users do not necessarily know each other by name, the access control setting is done based on the other users' properties (e.g., is over 18 or lives in Dresden). With the developed extension for the Open Source Mediawiki/phpBB forum software, the user as originator is able to specify access control policies for her contribution. The extensions modify and upgrade the original access control features of the wiki/forum, so that they work together with the PRIMEcore's (cf. §10) access control components. These components encompass:

- creating and editing access control policies,
- using anonymous credentials, and
- checking access control rights.

In a wiki/forum with such extended access control features, each user is allowed to specify which properties someone has to possess in order to access the user's contribution. Detailed information and instructions can be found in PrimeLife Deliverable D1.2.2 [Pri10].

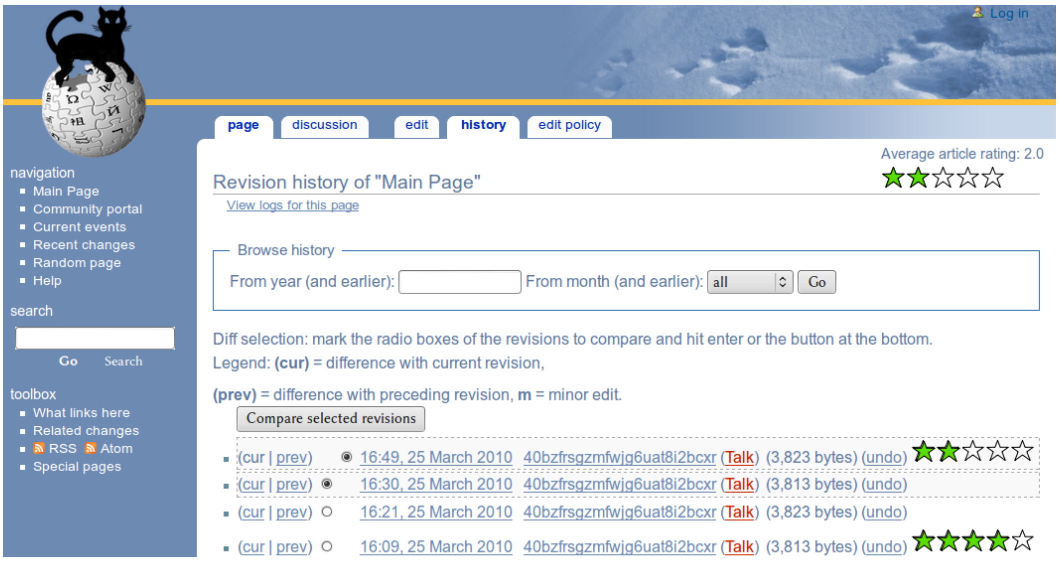
11.2 Impact

Both implementations are available via `www.primelife.eu/results/opensource` and target on two kinds of users. First, administrators of wiki and forum servers may use it to provide enhanced access control for their users. Second, as the software is released under free licences, developers of other applications can use the prototype as a reference implementation, how the PRIME core can be used.

12 MediaWiki – Reputation Extension

12.1 Description

The MediaWiki reputation extension provides functionality to handle the reputations of users and ratings of wiki pages. The extension is based on the ReaderFeedback extension[6] and requires the MediaWiki – Privacy enhanced access control extension to be installed (cf. §11). The extension provides a web form to rate the current revision for every content page. The average rating will be displayed at every page. Additionally, a special page displays the rating history of a page and average ratings are displayed in the page history (cf. Figure 5). Finally, the author's reputation value is updated using other users' ratings.



Average article rating: 2.0 ★★☆☆☆

Revision history of "Main Page"

View logs for this page

Browse history

From year (and earlier): From month (and earlier): all Go

Diff selection: mark the radio boxes of the revisions to compare and hit enter or the button at the bottom.
Legend: **(cur)** = difference with current revision,

(prev) = difference with preceding revision, **m** = minor edit.

Compare selected revisions

- (cur | prev) ● 16:49, 25 March 2010 40bzfrsgzmfwjg6uat8i2bcxr (Talk) (3,823 bytes) (undo) ★★☆☆☆
- (cur | prev) ● 16:30, 25 March 2010 40bzfrsgzmfwjg6uat8i2bcxr (Talk) (3,813 bytes) (undo)
- (cur | prev) ○ 16:21, 25 March 2010 40bzfrsgzmfwjg6uat8i2bcxr (Talk) (3,823 bytes) (undo)
- (cur | prev) ○ 16:09, 25 March 2010 40bzfrsgzmfwjg6uat8i2bcxr (Talk) (3,813 bytes) (undo) ★★★★☆

Figure 5: User Interface of MediaWiki History with the Reputation extension

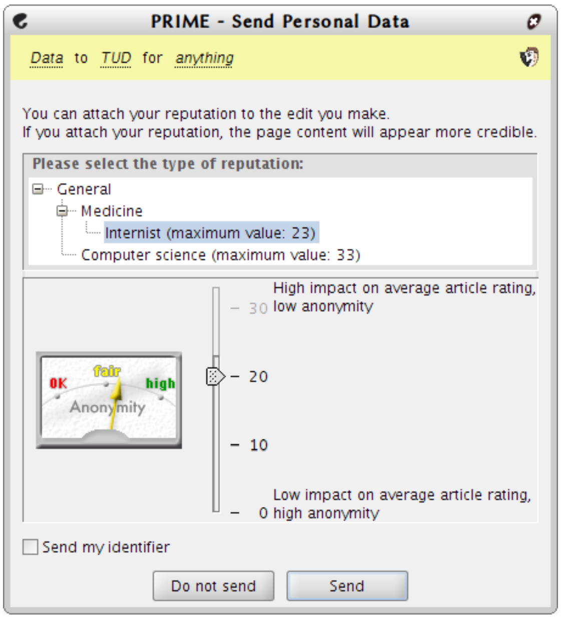
The reputation of authors as well as raters is stored in a privacy-friendly decentralized way. For issuing these reputation-credentials, the PRIME core is used (cf. §10).

To rate some page, the so called "Send Personal Data Dialog" is modified using the PRIME core plug-in mechanism. A screenshot of this dialog is displayed in Figure 6.

12.2 Impact

As the sourcecode modifies the ReaderFeedback extension, the improvements which were done within PrimeLife might flow back to the original extension. To our knowledge, we have been the first to implement the concept of a decentralized reputation storage and cryptographic proofs in a Web 2.0 reputation system. Our implementation is far from

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



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)

☐ Send my identifier

OK
fair
high
Anonymity
High impact on average article rating, low anonymity
30
20
10
0
Low impact on average article rating, high anonymity
Do not send
Send

Figure 6: User interface of the dialog asking for the user's reputation value

being stable and has to be improved before going into production code. However, the code is a good starting point for developers who want to implement such a system. Further, the code may be used as an example implementation, how to use the PRIMEcore as well as its plug-in mechanisms.

13 phpBB Privacy-Awareness Support: Personal Data MOD

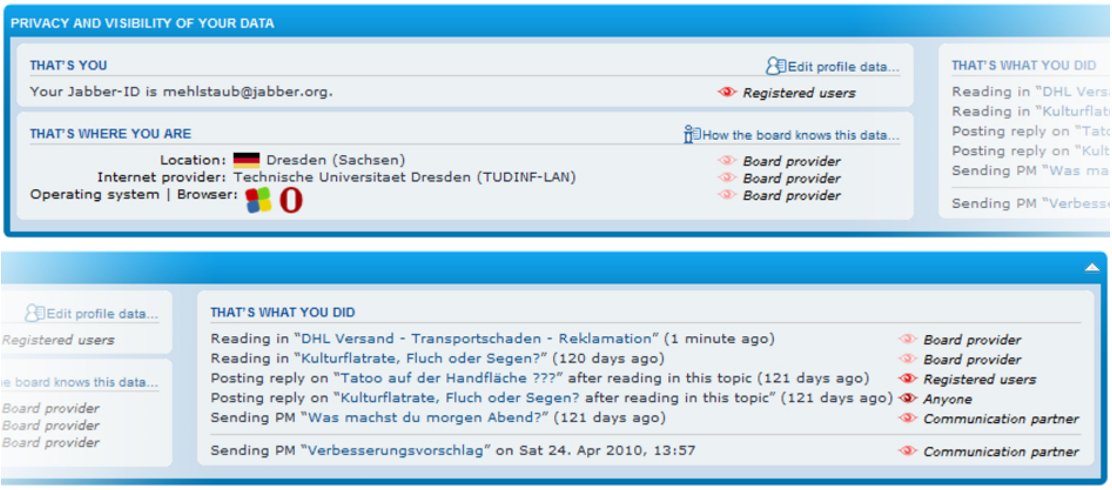
13.1 Description

When interacting with others on the Internet, users share a lot of personal data with a potentially large but "invisible" audience. There are tools available that give users feedback about their IP address, location, browser etc. and that can be integrated into websites (e.g., [Mof10]). However, showing users their IP address does not mean that they know what this means and who else may see this information. Therefore, we have developed a tool for the users' privacy awareness in a comprehensive way, i.e., a tool that informs them which explicitly and implicitly disclosed data are visible to whom. For concreteness, we have chosen to do this for the popular phpBB forum software, which is available with a copyleft license and is developed and supported by an Open Source community[7]. Thus, the objective of the *Personal Data MOD*[8] that we developed

[7] `http://www.phpbb.com/`

[8] MOD is the abbreviation for modification, a usual concept for extension of original phpBB software.

is to provide information about the visibility of personal data to phpBB forum users and thereby raising these users' privacy awareness.



PRIVACY AND VISIBILITY OF YOUR DATA

THAT'S YOU — Edit profile data...

Your Jabber-ID is mehlstaub@jabber.org. — Registered users

THAT'S WHERE YOU ARE — How the board knows this data...

Location: Dresden (Sachsen) — Board provider
Internet provider: Technische Universitaet Dresden (TUDINF-LAN) — Board provider
Operating system | Browser: — Board provider

THAT'S WHAT YOU DID
Reading in "DHL Vers
Reading in "Kulturflat
Posting reply on "Tato
Posting reply on "Kult
Sending PM "Was ma
Sending PM "Verbess

Edit profile data...
Registered users
e board knows this data...
Board provider
Board provider
Board provider

THAT'S WHAT YOU DID

Reading in "DHL Versand - Transportschaden - Reklamation" (1 minute ago) — Board provider
Reading in "Kulturflatrate, Fluch oder Segen?" (120 days ago) — Board provider
Posting reply on "Tatoo auf der Handfläche ???" after reading in this topic (121 days ago) — Registered users
Posting reply on "Kulturflatrate, Fluch oder Segen? after reading in this topic" (121 days ago) — Anyone
Sending PM "Was machst du morgen Abend?" (121 days ago) — Communication partner

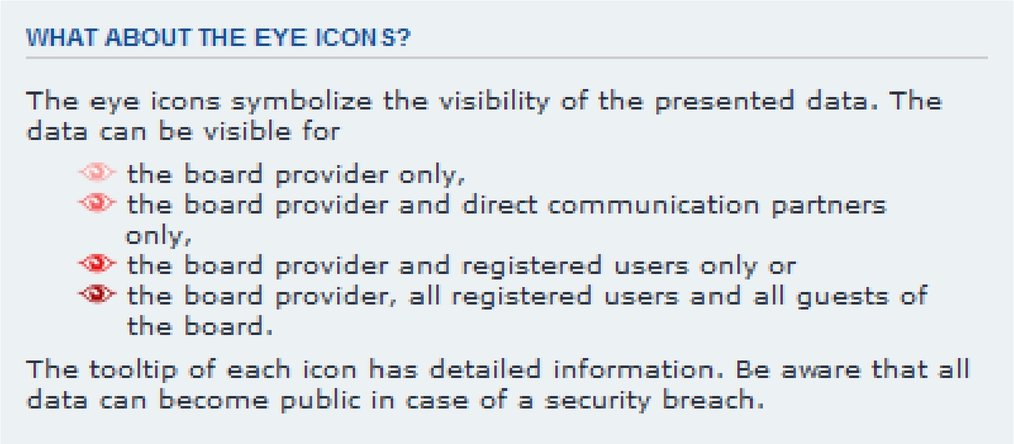
Sending PM "Verbesserungsvorschlag" on Sat 24. Apr 2010, 13:57 — Communication partner

Figure 7: User interface of Personal Data MOD.

Forum users get displayed Personal Data MOD on top of the forum (see Fig. 7). On the left side, a user is reminded about personal data from her profile and its visibility. The user is also informed about additional information which is automatically transmitted to the forum provider when visiting the forum. On the right side, a user is notified about the visibility of her latest actions and she also learns that the forum does not "forget" old actions, but that everything is logged and can be looked up even after a longer time period. Personal Data MOD distinguishes four visibility classes for user's personal data (see Fig. 8).

13.2 Impact

Personal Data MOD has two main target groups. First, all *providers* of phpBB forums can easily download Personal Data MOD from the project website [Pri] and use it with their installation of phpBB3. Second, all *end users* of phpBB forums with Personal Data MOD are constantly supported in making privacy-aware decisions whether or not to disclose personal data in the forum. The impact of Personal Data MOD on users' privacy awareness was confirmed by several empirical studies [Pöt10, PWG10]. To encourage use and potential further development of Personal Data MOD, it is released under GPLv2.



WHAT ABOUT THE EYE ICONS?

The eye icons symbolize the visibility of the presented data. The data can be visible for

- the board provider only,
- the board provider and direct communication partners only,
- the board provider and registered users only or
- the board provider, all registered users and all guests of the board.

The tooltip of each icon has detailed information. Be aware that all data can become public in case of a security breach.

Figure 8: Eye icons in different shades of red representing four visibility classes for personal data.

14 Conclusion

PrimeLife has published a large number of tools and mechanisms as open source and thereby ensured that its results will be usable after the project had ended. For surprisingly many of them we already know that they are or will be used by other existing or planned projects and have already helped to share and distribute the ideas and concepts developed by the project. We have thus all reason to believe that we have been very successful in making “privacy by design” a reality in the near future.

Bibliography

[CDF$^+$09] V. Ciriani, S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Keep a few: Outsourcing data while maintaining confidentiality. In *Proc. of the 14th European Symposium On Research In Computer Security (ESORICS 2009)*, Saint Malo, France, September 2009.

[CMN$^+$10] Jan Camenisch, Sebastian Mödersheim, Gregory Neven, Franz-Stefan Preiss, and Dieter Sommer. A card requirements language enabling privacy-preserving access control. In James B. D. Joshi and Barbara Carminati, editors, *SACMAT*, pages 119–128. ACM, 2010.

[Dan07] Luuk Danes. Smart card integration in the pseudonym system idemix. Master's thesis, University of Groningen, Mathematics Department, 2007.

[DFJ$^+$10a] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, G. Pelosi, and P. Samarati. Encryption-based policy enforcement for cloud storage. In *Proc. of the 1st ICDCS Workshop on Security and Privacy in Cloud Computing (SPCC 2010)*, Genova, Italy, June 2010.

[DFJ$^+$10b] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Encryption policies for regulating access to outsourced data. *ACM Transactions on Database Systems (TODS)*, 35(2):12:1–12:46, April 2010.

[Mof10] Jon Moffet. PHP Detector library. `http://phpcode.mypapit.net/demo/detector/detector.zip`, November 2010.

[Pöt10] Stefanie Pötzsch. Einfluss wahrgenommener privatsphäre und anonymität auf forennutzer. In Schroeder Ulrik, editor, *Interaktive Kulturen. Proceedings Mensch & Computer 2010*, Berlin, 2010. LogosVerlag.

[Pri] PrimeLife. phpbb privacy-awareness support: Personal data mod. `http://www.primelife.eu/results/opensource/59-phpbb-pam`. Last accessed 5 May 2011.

[Pri10] 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.

[PWG10] Stefanie Pötzsch, Peter Wolkerstorfer, and Cornelia Graf. Privacy-awareness information for web forums: results from an empirical study. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, NordiCHI '10, pages 363–372, New York, NY, USA, 2010. ACM.

[VVL$^+$10] Kristof Verslype, Pieter Verhaeghe, Jorn Lapon, Vincent Naessens, and Bart De Decker. Priman: a privacy-preserving identity framework. In *Data and Applications Security and Privacy XXIV*, volume 6166, pages 327–334. Springer, June 2010.