

Active Data Protection with Data Journals

Lars Brückner Jan Steffan Wesley Terpstra
Uwe Wilhelm
Darmstadt University of Technology
IT Transfer Office (ITO)
Wilhelminenstr. 7
64283 Darmstadt, Germany
{brueckner,steffan,terpstra}@ito.tu-darmstadt.de
T-Systems Technologiezentrum
64307 Darmstadt, Germany
Uwe.Wilhelm@T-Systems.com

Abstract: A major privacy problem on the internet is the unrestricted sharing of user data between services and other parties. The EU privacy legislation grants the user the rights to restrict the dissemination of his personal data, but users often simply do not know which kind of data was collected by these services. In this paper we propose data journals as a new kind of privacy enhancement technology to increase the user's ability to take advantage of his rights. A data journal is a tool that records the disclosure of personal data to services and collects related information about the service provider's identity and its privacy policy. We describe how data journals work, how the user can benefit from their usage, and their relation to other privacy enhancement technologies. Two prototype implementations show that data journals can be implemented on without changes to existing services or big changes of the user's browsing experience.

1 Introduction

Privacy problems are a major threat to the developing information society. Services collect a lot of personal data about their users. While a lot of this data is necessary to actually fulfill the services' purpose, some of the data is merely collected by the service provider to obtain more information about their users. This can have positive effects for the users and is acceptable if the user is given a choice to supply more information than what is strictly necessary to provide the service, but often the user is not given this choice. The problem of collection of user data is further aggravated by the fact that this data is often shared with other parties, e.g. for marketing purposes.

From a technical point of view, the data that services collect about their users can be divided in two categories: network-related and person-related. Network-related data includes all kind of information that is derived from network traffic. This includes cookies, IP-address, and HTTP-headers. Person-related data includes data about the human indi-

vidual. This includes the name, address, billing information, and all kinds of demographic data (age, interests, hobbies, etc).

With regard to privacy, the extensive collection and analysis of network-related data is a common technique to build user profiles that track the user's behaviour across multiple sessions and different services. Most users are not aware these practices. The situation is different for person-related data, especially for contact and billing information. Users enter it manually. They are aware that this information identifies themselves and that it is often passed to other entities; this is not different from the handling of these kinds of data in the real world. There have been a lot of press reports on the lack of privacy on the internet; most of them deal with the abuse of credit card or other identity information, e.g. spam mail. This has led to wide-spread distrust among users which can be observed in the well-accepted assumption that the lack of trust in e-commerce services is the reason for their slow acceptance (a poll by Business Week [Ham98], has found that 61% of poll respondents say they would use the Internet more if their privacy would be protected, see also [Sau01]).

The European Union's privacy regime [Com95] grants the right of informational self-determination to all users of electronic services. This includes the user's right to retain some control over the use and further dissemination of the information that he has disclosed to services. Service providers have to inform the user about the kind of information that they collect about him and for what purpose the information is used. The user can query service providers for the information they have stored about him, revoke his consent, demand corrections, and demand the removal of his data if it is no longer required.

We observed that most available privacy enhancement tools concentrate on anonymizing network-related data and that they provide adequate solutions for the problem of user profiling based on it. This includes systems for anonymous web browsing (e.g. [JAP]) and cookie blockers (e.g. [Pri]). The disclosure of person-related information and its later management is out of the scope of these tools. However, given the lack of anonymous payment or delivery systems, virtually all e-commerce sites require personal identification. There have been several attempts to establish infomediary services that manage the user's personal data. The most prominent of these services is Passport [Mic], but a lot of similar services have been proposed [Cra99]. The features of these infomediaries are very similar: The user can register one or more profiles containing a set of his personal information, and specify which profile is shared with a specific service. So far none of these infomediaries has achieved wide acceptance. The reason for this is twofold: Firstly, users regarded the infomediaries themselves as a privacy threat because they collect a lot of private information, can keep track of user transactions, and have the intent to increase the sharing of data. On the other hand, from a service provider's point of view, the infomediaries try to take control of an important part of the service provider's operations; some infomediaries require that services install extra software and pay royalties to participate.

The lack of tools to handle the management of his disclosed personal data puts a heavy burden on the user. Although the laws that grant the user the right of informational self-determination are in place, alone they are ineffective. The fundamental problem is that the fine grained control granted to individuals by law requires that they know which entities have their information. Recording this manually is too much of a burden for most individ-

uals, particularly because they will want to exercise their control over their information at a much later time than the time of disclosure.

This paper describes our approach to support the user in turning the usage of his rights on informational self-determination into an active part of his internet activities. We propose the concept of data journals, which keep track of the circumstances under which sensitive information was disclosed by a user. We will show that data journals provide a real benefit to the user and that they can be implemented without changes to existing web services.

This remainder of this paper is structured as follows: We further elaborate our idea of a data journal in section 2 and develop requirements a data journal should meet to become a useful tool for end users. Then we describe our two prototype implementations of a data journal and how they meet these requirements. The first prototype is based on an HTTP proxy (section 3), while the second is implemented as a component for the Mozilla web browser (section 4). In section 5, related work, we compare our approach to other privacy enhancing technologies and standards. We conclude the paper and give an outlook on future work in section 6.

2 General Approach

The primary goal of a data journal is to free the user from the tedious task of bookkeeping about the personal information that he has given to services. If sensitive information is disclosed, the journal gathers additional information about the service provider. The data journal should also provide means add unique marks to otherwise identical data that is disclosed to different services. This enables the user to contact the service provider at a later time and demand information, updates or the removal of his data.

This tool puts the user in a position where he can actively exercise the rights that are part of the informational self-determination. We can take advantage of P3P [CLM⁺02], which is a concept that allows a service to state its privacy practises, so that a prospective client can decide whether he wants to do business with this service. The slow adoption of P3P can be seen as a sign that something is missing. We believe that one of the missing features is to make the P3P policy under which a particular interaction has taken place accessible to the user in the case of an actual dispute. Other tasks that should be supported by data journals are to find out which services know a particular e-mail address, the notification of service that an address has changed, to check if a service is really authorized to send a newsletter, or to demand the removal of data from a service's databases.

To handle such tasks, the user needs detailed information: what kind of data was disclosed, when did it happen, what conditions about the data usage where stated, and how to contact the service provider or other responsible parties. Typically the user has to perform these tasks at a much later point in time than his last interaction with the service. Due to the fast change of the internet sites, a lot of the required information (e.g. privacy policies) may not be available online when it is required for management. For that reason the data journal can not depend on other services performing its tasks. Based on our idea of management tasks and our analysis of other approaches we defined the following requirements:

1. Record the type of information that has been disclosed, and what usage practices were stated
2. Record contact and other legally relevant information on the collecting entity
3. Work with existing internet infrastructure and services
4. Never prevent the user from doing anything he can do today
5. Never require parties who do not benefit to run a component
6. Protect the recorded information
7. Easy to install and use
8. Be self-contained and small

Items 1 and 2 address the core functionality that have been described above. In addition to these two requirements that define the function of data journals, we have identified requirements that are more concerned about the context in which data journals can be successful. No new infrastructure can be required. We consider it highly unlikely that both clients and servers will adopt a technology which is not useful without the other component. Even large companies like Microsoft have difficulty in forcing this form of adoption. If two components are highly useful on their own, and provide additional benefit during interaction, this is acceptable. However, the core benefit of our applications must never rely on this interaction. This is the reasoning behind requirements 3 and 5. We do not want to prevent the user from getting his work done. If our technologies prevented the correct function of services that the user needed, then the user would likely not use the data journal. For example, if a web site does not have a privacy policy, the tool should not prevent the user from viewing this site. However, the tool might warn the user and allow him to cancel his submission if he submits data and the conditions appear dangerous. This is requirement 4. The data journal gathers a lot of information about the user's e-commerce transactions. Therefore it might be a valuable target for attackers; it must be protected against unauthorized access, manipulation, and other types of attacks. An inadequate solution might pose a bigger threat to the user's privacy than the problems the journal tries to solve. This is requirement 6.

The specific goal of this paper is to describe approaches that demonstrate that these requirements can be met. We have two implementations which show that these requirements can be met.

3 Using an HTTP Proxy

The PRIMA Datamanager is a modified HTTP-proxy. It is implemented in Java using the Jigsaw [L⁺], Cocoon [Fou], and eXist [Mei] projects and was primarily a prototype intended as a first proof-of-concept.

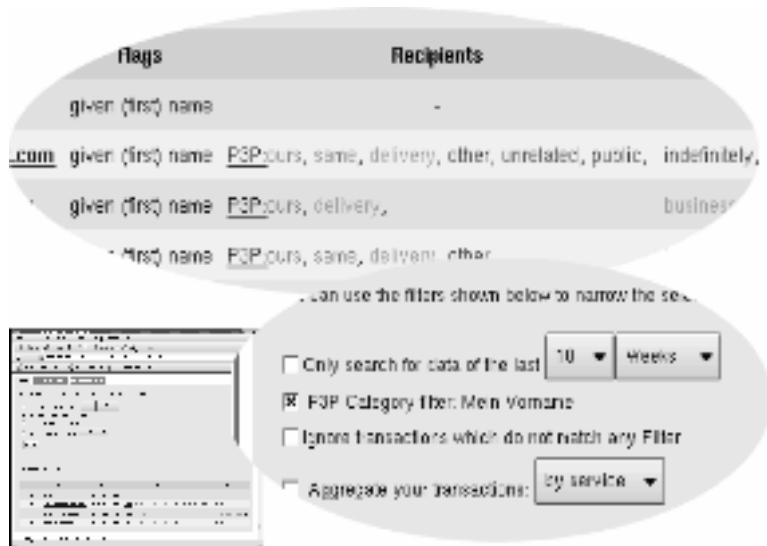


Figure 1: Viewing the Journal

3.1 Meeting the requirements

Proxies are an established technology and supported by nearly every existing web browser. Therefore, this solution meets requirement 3 for the case of normal unencrypted HTTP-traffic. Because the proxy is placed between the web site and the browser, it can analyze all the data being transferred back and forth. This puts it in an excellent position to fulfill requirement 1. To fulfill requirement 2, the Datamanager uses any P3P policy that the web site may provide. This information is a rich source of not only contact information, but also of what the collecting entity intends to do with the data. The Datamanager summarizes this information for the user in a table as shown in figure 1. This can help the user in deciding whether or not he wants to revoke his consent sooner rather than later. It can also let the user know if the collected data should have already been deleted in accordance with the expiry limits in the P3P document.

Unfortunately, not many web sites provide a P3P policy. This means that this implementation only satisfies requirement 2 for a few web sites. Furthermore, as there is an ongoing debate whether P3P documents are legally binding, and are provided by the data collecting entity, they are hardly a trust-worthy data-source. If web sites which collect information were legally required to provide an accurate P3P policy, then both objections above would be met. This was our hope when the Datamanager was written. It is unlikely that our prototype will meet requirement 2 for most use cases.

Requirements 4, 5, and 8 are met. Since the proxy merely passively records traffic, it never prevents the user from doing anything. The proxy is installed in only one location and requires no support from either web browser authors or web servers. Finally, the proxy is a single component and quite small.

The detection of the information is done by configuring string matching rules. Although this can be done subsequently, entering samples of all personal data is a tedious task that might not be accepted by many users. In this point, requirement 7 is not satisfactorily fulfilled. One way of resolving this problem is shown in the browser based approach described in section 4.

In order to allow ex post categorization of data, the Datamanager records the tuple (time, URL, disclosed data, p3p policy) for each out-bound request. This means that it records the actual private information submitted in its own log. Without a way to safeguard this data this is a severe risk if this file could be stolen. In order to meet requirement 6 an encryption mechanism would have to be added.

One benefit of the proxy is that it can be executed on a different machine than the browser itself. This capability allows a user to have a common log, even if he uses browsers from different machines. One drawback of proxy approach is that the configuration of proxies is not a simple task. In particular when a user attempts to chain several proxies this approach does not meet requirement 7.

The main difficulty with the proxy is in accessing data that is transmitted within an SSL encrypted connection. As SSL is intended to provide an encrypted end-to-end communication channel between browser and web site, there is no way for a proxy to access transmitted data in clear-text without breaking the end-to-end security paradigm. From a technical point of view, it would be possible to implement a proxy that acts as a man-in-the-middle by establishing separate SSL connections with the browser and the web site. Even if the user would fully trust the Datamanager and accept the interruption of the end-to-end SSL connection, this approach would present an incorrect SSL certificate and could cause other client-side security logic to fail. Omitting the detection of data submitted through encrypted connections isn't an option either as SSL is used for the transmission of highly sensitive private information in particular. This means that our proxy based implementation cannot simultaneously satisfy requirements 1 and 3 for web sites using SSL.

3.2 Conclusion

With the implementation of the PRIMA Datamanager we verified that our approach of collecting and classifying submitted private information in a journal is feasible. We identified some critical issues such as the need to further minimize the user's effort for setup and the protection of the data contained in the journal itself. We realized that P3P alone is not sufficient to collect information about the recipients of private data. We also studied the advantages and disadvantages of locating the recording component inside a proxy and found that the issue of encrypted connections can not be resolved and is serious enough to outweigh the benefits.



Figure 2: Viewing the iJournal

4 Extending a Web-browser

Based on our experience from the first prototype, we implemented a second prototype as a web browser component. The browser we chose to target was Mozilla [Org] since it is mature, cross-platform, open-source, and has a large development community.

4.1 Meeting the requirements

This approach will work with any web site. However, it will only work with the Mozilla web browser. For users of the Mozilla web browser, it certainly meets requirement 3. However, for other users, it could be said to fail this point. Because Mozilla is cross-platform, users who value their privacy always have the option of upgrading to Mozilla regardless of the operating system they run. Therefore, we choose to say that requirement 3 is met. Certainly support for more browsers would be better, but current cross-browser plugin technology is not powerful enough to make this possible. Since the iJournal lives inside the web browser, it is in as good a position as the Datamanager to observe what

information is being disclosed. Therefore, requirement 1 is met.

Where the Datamanager gathered the P3P policy during the submission of data, the iJournal too takes out-of-band action at submission time. It grabs records from the whois database for the domain and IP address. It also examines the SSL certificate of the web site and records the P3P policy. Therefore, the tool meets requirement 2. Unlike the Datamanager, the iJournal does not trust information from the P3P policy. Instead, it relies more heavily on third-party verified information from the whois database [HSF85] and SSL certificate. Furthermore, since whois information is available for every web site and SSL certificates are available for every secure connection, the iJournal nearly always succeeds in requirement 2.¹

Requirements 4, 5 and 8 are met. The component never prevents access, and it does not change the browsers interaction with the webserver. Only the benefiting user must install it. A browser component is also much simpler to install than a separate proxy. There is no problem with proxy chaining, configuration, or even installing another program. Simply clicking an installation hyper-link on a web-page, confirming the install, and restarting the browser is sufficient to have a fully functioning setup of the iJournal. In the future, we may add a wizard to help ensure that important properties such as the user's name are already known by the browser in case Mozilla's automatic mechanisms fail. At any rate, requirement 7 is met.

The iJournal uses RDF[LS99], to record meta-information about the user's transactions. For example, it records facts like "Bookshop knows my name" and "My name is known by Bookshop". These sorts of facts do not include what the user's name is, thus minimising the impact if the data-source is compromised. The meta-information can also include details which describe the logical relations and let the user browse the information in a convenient manner as shown in figure 2.

Mozilla already includes a component for automatic form submission, the *wallet*. It tries to infer the user's name, and other pieces of information. It also already encrypts the user's personal data. Therefore, by storing only facts of the form "Bookshop.com knows my name" and relying on the existing wallet, requirement 6 is met in a fairly satisfactory manner. Because the user has a simple interface for viewing the information as shown in figure 2 and does not need to enter information on an item by item basis, the tool satisfies requirement 7.

4.2 Information Permutation

Since the IJournal runs inside the browser, it is also a trivial matter to slightly permute the user's data to assist in determining where the data came from. One feature of the IJournal is to add some magic salt to fields. For example, "John Doe" is changed to "John WT Doe". These options are configurable, and if a user has control over his email routing he can

¹Obviously the IJournal cannot guarantee a secure identification and provide a proof in the legal sense. The quality of the whois database differs between the internet top-level domains, it may not be up to date. The same is true for SSL Certificates issued by different CAs. Also, whois and SSL certificate may be forged.

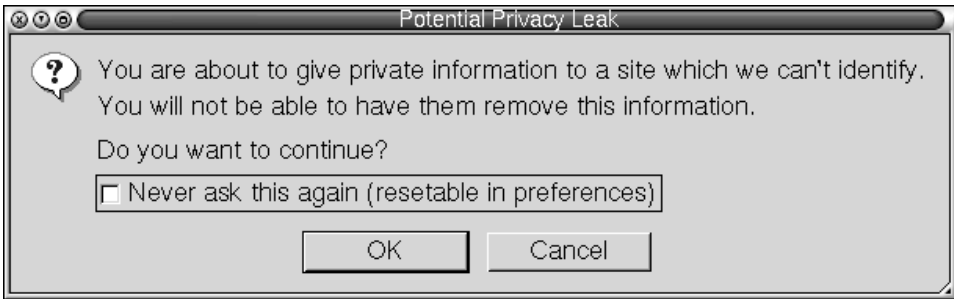


Figure 3: Something suspicious

also transform his email address like “johndoe@wt.mydomain.com”, “wt-johndoe@myisp.com”, etc. This way, if someone ever contacts the user, he can immediately ask the iJournal “which company was given a salt of WT”.

To do the actual pattern matching, the tool uses an enhanced version of the edit-distance algorithm. This version of the edit-distance algorithm is very good at inferring that “John James Doe” could be entered as “John J Doe”, “John D”, “Jhon Jaems Doe”, etc. In this manner the tool can accommodate slight permutations of the data due to typos and abbreviations.

4.3 Protect The User When Possible

First, the iJournal checks the submitted data for personal information. If personal information is present, it will gather information about the collecting entity. If the iJournal detects that something suspicious is happening, it will warn the user at this time, before the data is submitted. The user may then cancel the form submission as illustrated in figure 3. The iJournal presently considers a form submission suspicious if it is unable to acquire two pieces of contact information from a trusted third party.

4.4 Conclusions

The most important improvement of the second approach is that it solves problem with encrypted connections. Also, the integration offers a much better user interface and the option of user interaction without changing the browser’s interaction with the webserver. We could also reuse existing components of the browser like the wallet. The major drawback of this approach is obvious. The iJournal only works with Mozilla and some Mozilla derivatives. The coupling required is also so high that changing Mozilla versions can break the component. If the user accesses the web from different computers, he has to take care of synchronizing the journals.

5 Related Work

As mentioned above, most available privacy enhancing technologies focus on anonymizing personal information. However, services often require real personal information (e.g. for billing and goods delivery) to fulfill their purpose, and therefore higher-level concepts and technologies are needed. Identity managers take a different approach on the handling of the user's data by using predefined profiles. The W3C technical recommendation P3P sets a standard for describing privacy policies about person- and network-related data, but P3P's concept of a user agent is limited to avoid data disclosure at the expense of not using services.

The idea of the Identity Manager as described in [CK01], follows a similar idea as the idea of the data journal described in this paper. An Identity Manager also has to know what information it is allowed to disclose to which entity, but it requires the user to think about this beforehand (the user has to select the identity under which he wants to conduct a particular interaction). As such, the Identity Manager can be seen as a top down approach, while the data journal presented here is rather a bottom up approach. The user does not have to decide on a disclosure policy beforehand but can simply continue with his regular interactions, which are logged in his journal. In the case when he wants to resolve a problem he has to invest more effort in pinpointing the actual source of the problem, since the data gathered in his journal is less structured.

The Platform for Privacy Preferences P3P [CLM⁺02] was developed to enable the automatic exchange and negotiation of privacy policies between users and services. The main feature is a XML vocabulary to describe privacy policies in terms of data types, collecting purpose, retention period, and other parties the service might share the data with. A user agent is expected to collect a service's privacy policy, see if it matches the user's preferences, and alert the user or block access to the service if the policy does not match his preferences. There is one reference implementation of a P3P user agent [HJW02] that shows several technical shortages of the current standard. It is implemented as a http proxy. Another example of a P3P user agent is the Privacy Bird [Res]. Privacy Bird is implemented as a plugin for the Microsoft Internet Explorer and therefore has a much better UI integration. Unlike a full P3P User Agent, it does not block access to sites without a suitable P3P policy, but gives an audiovisual warning. Also, if a user submits any manually entered string to a service without a suitable P3P policy a warning is presented to the user. Unlike the IJournal, Privacy Bird does not analyze the submitted data if it really contains private information and does not provide any means to log data transfers. Current commercial browsers from Netscape and Microsoft only support a very small subset of P3P, allowing the user to change the cookie acceptance of the browser based on the P3P policy of the server. The slow adoption of P3P on the server side renders the existing P3P user agents very much ineffective for the users. There is an ongoing debate about P3P being accepted by legal authorities as a way to express and manage data protection on the net.

As our idea of a data journal does not require changes to the existing net infrastructure, the data journal cannot guarantee that service providers stick to their stated policies. Its mechanisms for identifying malicious service providers can be circumvented. Data Licences

[CY] are an approach to establish a more trustworthy infrastructure. Personal information is accompanied with a license issued by the user describing his consent how the information can be used by the service.

6 Conclusion

In this paper we have discussed the problem of disclosing personal information to service providers, which is a prerequisite for the usage of many web services. Our proposal for addressing this problem is to empower the user with a technological tool that allows him to actually take advantage of existing privacy legislation.

The approach that we have chosen for this empowerment is to free the user from the tedious task of bookkeeping about the various pieces of personal information that are disclosed to different services. If sensitive information is disclosed, the journal marks that information and gathers additional information about the service provider and the service's stated privacy practises. This enables the user to contact the service provider at a later time and demand information, updates, or the removal of his data. We have argued that data journals cover an aspect of privacy that is not handled by current tools.

We described two prototypes that we have implemented in order to verify our approach. We have shown that data journals provide a real benefit for the user. They can be implemented without changes to existing web services, and therefore work without support from service providers.

In the future, we want to continue the further developement of the iJournal. We need to further investigate its security und usability issues. Also, our current implementation can only detect contact and billing information that is entered via the keyboard. It would be desirable to detect more types of information (e.g. hobbies and interests) and different kinds of input, but this would require a complex analysis of the HTML structure. Therefore, we see the need to integrate our approach with other privacy enhancement technologies, such as an identity manager, into a consistent architecture.

Acknowledgements

We would like to thank Axel Hecht and Netscape employees Chris Waterson, Peter Van der Beken, and Jan Varga for helping us to understand the internals of the Mozilla browser.

References

- [CK01] Sebastian Clauß and Marit Köhntopp. Identity Management and Its Support of multilateral Security. *Computer Networks*, 37:205–219, 2001.
- [CLM⁺02] Lorrie Cranor, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, and Joseph Reagle. *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification*. W3C, 2002. <http://www.w3.org/TR/2002/REC-P3P-20020416/>.

- [Com95] European Community. *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*, October 1995. Homepage: <http://www2.echo.lu/legal/en/dataprot/directiv/directiv.html>.
- [Cra99] Lorrie Faith Cranor. Agents of Choice: Tools that Facilitate Notice and Choice about Web Site Data Practices. In *Proceedings of the 21st International Conference on Privacy and Personal Data Protection, 13–15 September 1999, Hong Kong SAR, China*, pages 19–25, 1999.
- [CY] Shi-Cho Cha and Yuh-Jzer Young. From P3P to Data Licenses. To appear in: Proceedings of the Third Workshop on Privacy Enhancing Technologies (PET2003) Pre-Proceedings available online at <http://www.petorkshop.org>.
- [Fou] Apache Software Foundation. Cocoon XML Publishing Framework 2. <http://java.apache.org/cocoon2>.
- [Ham98] K. H. Hammonds. Online Insecurity (Harris Poll). *Business Week*, March 1998.
- [HJW02] Giles Hogben, Tom Jackson, and Marc Wilikens. A Fully Compliant Research Implementation of the P3P Standard for Privacy Protection: Experiences and Recommendations. In *Computer Security – ESORICS 2002: 7th European Symposium on Research in Computer Security, Zurich, Switzerland, Proceedings*, volume 2502 of *Lecture Notes in Computer Science*, pages 104–125, 2002.
- [HSF85] K. Harrenstien, M. Stahl, and E. Feinler. *NICNAME/WHOIS RFC 954*, October 1985. <http://rfc.sunsite.dk/rfc/rfc954.html>.
- [JAP] JAP Anonymity and Privacy Proxy. <http://anon.inf.tu-dresden.de/>.
- [L⁺] Yves Lafon et al. Jigsaw Webserver. <http://www.w3c.org/Jigsaw>.
- [LS99] Ora Lassila and Ralph R. Swick. *Resource Description Framework (RDF) Model and Syntax Specification*. W3C, 1999. <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222>.
- [Mei] Wolfgang Meier. eXist XML Database. <http://exist.sourceforge.net>.
- [Mic] Microsoft. Microsoft Passport. <http://www.passport.net>.
- [Org] The Mozilla Organization. The Mozilla Web Browser. <http://www.mozilla.org/projects/seamoney/>.
- [Pri] Privoxy Project. <http://www.privoxy.org>.
- [Res] AT&T Research. Privacy Bird. <http://www.privacybird.com>.
- [Sau01] Christopher Saunders. Web-Savvy Consumers Wary of Data Loss. *Internet News*, June 2001. Online at http://www.internetnews.com/IAR/article.php/12_781291.