Personal Federation Control with the Identity Dashboard

Jonathan Scudder, Audun Jøsang

Personal Federation Control with the Identity Dashboard

Jonathan Scudder[1] and Audun Jøsang[2]

[1] Department of Informatics, University of Oslo, Norway. jonathas@ifi.uio.no
[2] UniK Graduate Center, University of Oslo, Norway. josang@unik.no

Abstract. Current federated identity management solutions for open networks do not solve the scalability problems for users. In some cases, federation might even increase the identity management complexity that users need to handle. Solutions should empower users to actively participate in making decisions about their identity, but this is far from the current situation. This paper proposes the *Identity Dashboard* as a user-centric control component, providing users with tools they need to effectively partake in managing their own identities.

Keywords: Identity management, federation, SSO, single sign on, usability, security, authentication, SAML, privacy, privacy enhancing technologies

1 Introduction

Digital identities represent who we are when engaging in online activities and transactions. The rapid growth in the number of online services leads to an increasing number of different identities that each user needs to manage. At the same time, there is a proliferation of sensitive information on the Internet as the volume of personal details required by systems and sites is increasing.

Identity federation reduces this burden to a certain degree by addressing the transport of identity assertions [1] between

service providers and identity providers. Identity federation is strictly standards-based to allow interoperability between autonomous parties. These standards describe authentication-related messages and the ways in which they are passed between federated entities in order to allow a relying party to trust that a user has identified themselves to an asserting party. The two most widespread standards are SAML [1] and WS-Federation [2]. SAML 2.0 is a convergence of the Liberty ID-FF [3], Internet2 Shibboleth [4], and earlier SAML standards.

Other more recent developments include "user-centric" approaches to federation, such as Cardspace [12] and Higgins [13]. User-centric approaches have a different paradigm for handling user attributes which utilize client-based storage and authorization [14, 15]. In contrast to network-based federations where the personal identity information is not stored locally, user-centric approaches are not subject to concerns about who is in control of the data, which is the main motivation for this paper.

Identity federation opens the door to a win-win situation whereby the user experience is simpler, no longer requiring the user to log on to accounts individually, and at the same time more secure since user credentials can be better protected. For example, if a user only needs one password where they previously needed 10 passwords, then the new password can conceivably be longer, and changed more often without overtaxing the user. Due to the degree of trust involved in creating identity federations, legal and organisational requirements are also paramount, and are partially addressed by the prevailing standards.

A major challenge to federation is realising user-control. A user should ideally know which services they can access, and which attributes are being shared between identity providers and service providers. This is difficult to achieve in practice since the required information is not necessarily collected at a single point. An identity provider may know which attributes it is willing to make available to service providers, but only a subset of these may be requested by the service provider, and this may be controlled dynamically at runtime.

Another challenge to federation is that the degree of trust required does not foster truly large-scale federations. Thus the dream of a single sign-on where identity is authenticated only once for virtually all systems will not likely be achieved with the current frameworks alone. This leads to a future where users will have relationships to an increasing number of federations. The emergence of "super federations" which connect individual federations in inter-trust relationships leads to a less clear picture of how the user's identity is handled, and where identity attributes are being shared.

The development of identity federation technologies was inspired by the need to improve the user experience when accessing online services. Ironically, the technology-centric approach followed is now causing new usability problems that the SAML 2.0 and WS-Federation specifications do not account for.

This paper examines the need for user control in a SAML 2.0 federated environment, and proposes requirements for a new component called the Identity Dashboard in order to provide the user with a central point of contact through which they can effectively manage their federated identities. The Identity Dashboard is envisaged realised through extensions to the OASIS SAML 2.0 standard based on the requirements discovered.

Note that privacy *policies* have been explored elsewhere [11, 5, 6]. The Identity Dashboard addresses the problem of monitoring; having expressed the wish that an attribute should not be shared or used as an identifier, for example, has limited value if there is no way to monitor the usage.

2 Models for Identity Federation

Federated identity models are based on groups of *service providers* (SPs) that enter into a mutual security and authentication agreement in order to allow user SSO to their services. In the terminology of the Liberty Alliance, these groups are called *circles of trust* [3]. The user identity is asserted by one or more *identity providers* (IDPs). Identity federation can thus be defined as a set

of agreements, standards and technologies that enable SPs to recognise user identities and entitlements from IDPs [7].

There are two particular approaches to organizing federations as well as scenarios where identity federation are combined. These will be discussed briefly in order to explore the need for user control in these scenarios.

2.1 The hub-and-spoke model

The hub-and-spoke model for identity federation is where one or more centralized identity providers are federated with multiple separate service providers. This federation model is commonly found in environments with a degree of common management (e.g. governmental federations or umbrella companies). Note that service providers have no direct contact with each other within the federation context.

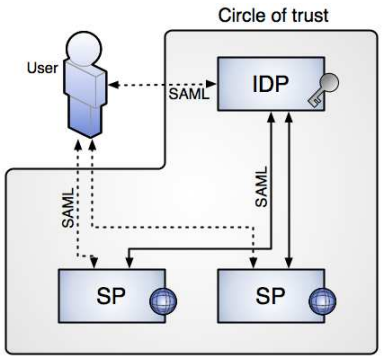

Fig. 1. Hub-and-spoke federation

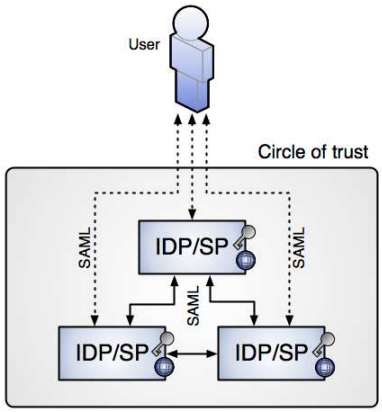

Fig. 2. Dual-role SP-IDPs

With a typically low ratio of IDPs to SPs, the number and variation of assertions being transferred are quite limited. Users can potentially manage their federated identity by interacting with the central IDP(s), and therefore maintain an overview. Note that IDPs *can* also perform as SPs, although this is not often the case, due to mercantile reasons[3].

With x IDPs and y SPs, the user must manage up to $x \times y$ attribute combinations and account links. Whilst this may be challenging, it is at least feasible for such a federation to provide users with sufficient information. Note that the SAML 2.0 specification does not include any standard protocol or requirement for the IDP to make federation information available to the user [1].

[3] An IDP must often provide a high level of availability, support services, and test labs. The costs for this may be passed on to federated SPs in some way, which underlines the need to avoid mixing IDP and SP roles withing the same deployment

2.2 The SP-IDP model

In the combined SP-IDP model, each SP also acts as an IDP, managing the name space of its own users whilst federating with the other SPs. Various silo identity domains are then linked together to form a federated domain [7], as shown in fig. 2. For authentication purposes, an SP will accept a *security assertion* [1] from another SP claiming that a given user has already been authenticated.

The ratio of IDPs to SPs in this model approaches 1:1. For each IDP in the circle of trust, there exists a set of attributes that can potentially be shared with other SPs. Furthermore, each IDP-SP relationship may be limited to a subset of the available attributes. This gives up to $x!$ attribute combinations and account links for the user to manage, where x is the number of dual-role SP-IDPs. Also, the lack of a single point of control[4] within the circle of trust prevents the user from effectively being able to manage or view these attribute combinations without visiting each and every IDP.

In such a model, it is difficult for the user to achieve a satisfactory overview of their federated identities. With increasing focus on privacy issues it is expected that privacy aspects of dual-role SP-IDP federations will have to be addressed in the near future.

2.3 Combining identity federations

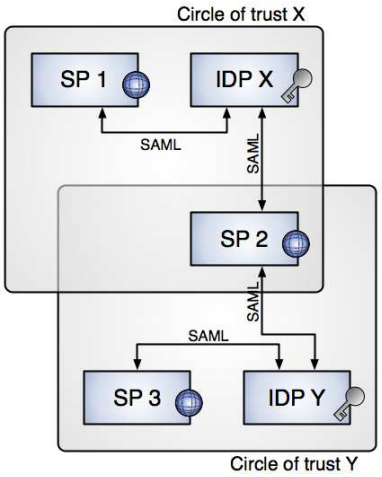

Fig. 3. Multiple circles of trust

[4] There could well be more than one point of control (IDP) in a centralized model, so a single point of control is not necessarily a challenge restricted to dual-role SP-IDP federations

Identity federations involve creating resonably tight trust and legal relationships. This limits the size of identity federations, and gives rise to the need for multiple federations. Whilst one may first address the need for informing and involving users *within* a federation, it is not particularly feasible for a user to have to check a multitude of Identity Dashboards in order to gain an overview of their federated identities. This gives rise to the notion of a cross-federation Identity Dashboard, capable of displaying information about many of the user's federations.

Multiple identity federations can also increase the need for the user to understand which attributes are being shared. For example, it is possible for a service provider to participate in two separate circles of trust (fig 3). If the user is presented with a choice of IDPs to authenticate to, information about which attributes would be shared may affect their decision.

Not only would the user have to relate to two different sources of information about their federated identity (IDP X and IDP Y), the user may also have trust issues based on which other SPs are involved in the respective circles of trust. For example, if a user does not trust the service provider SP 3 then they may not want to authenticate to IDP Y at all.

3 Are users interested?

Before looking closer at what should be controlled, the question of whether a user is interested in how their identity attributes are handled should be addressed. A survey conducted by Ackerman et al. [10] asked American internet users how important it would be for them to be notified of privacy policy conflicts for different types of information or action (Table 1).

> If you could configure your web browser to look for privacy policies and privacy seals of approval on web sites and let you know when you were visiting a site whose privacy practices might not be acceptable to you, which criteria would be most important to you? For each of the items in the left-most column, please indicate whether it is very important, somewhat important, or not important. [10]

In this survey, a very significant proportion of users were interested in the sharing of information (79%), whether the information is identifiable (75%), and knowing the kind of information being handled (69%); all of these are relevant for a federated identity model.

In a separate survey by Earp et al [16], respondents were asked to state their agreement or disagreement with statements grouped into six categories of personal identity information events. These results were then ranked according to importance (Table 2). Also here, the transfer (sharing) of information and awareness of the information was considered more important that being able to alter or personalize it.

Table 1. Privacy Survey Results Ackerman et al. 1999

Information	Very important
Sharing of information	79%
Identifiable use of information	75%
Purpose of information collected	74%
Mailing list removal upon request	74%
Kind of information	69%
Access to stored information	65%
Site run by trusted company	62%
Posts privacy policy	49%
Privacy seal of approval	39%
Disclosure of data retention policy	32%

Table 2. Privacy Survey Results Earp et al. 2005

Category	Importance (rank)
Transfer	1
Notice/Awareness	2
Storage	3
Participation/Access	4
Collection	5
Personalization	6

Both surveys find that the sharing of information is relatively important when compared with other uses and aspects of information privacy. The Identity Dashboard is accordingly focused on raising user awareness of the personal identity information being *shared.*

4 What a user needs to control

Identity federation involves sharing information across organizational borders. This information can include persistent identity attributes such as email addresses or social security numbers, and transient authentication information such as when and how you authenticated.

A typical federated authentication sequence based on SAML 2.0 is shown in figure 4 (redirects are not shown for the sake of simplicity). The following steps are involved[5]:

[5] Assumes persistent federation in a service provider initiated single sign on scenario

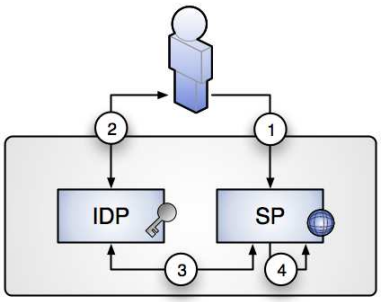

Fig. 4. Federated authentication

1. User attempts to access a service at SP site and is redirected to the appropriate IDP
2. IDP challenges the user to authenticate and redirects back to the SP
3. IDP issues a SAML assertion to the SP containing a unique identifier and user attributes
4. SP trusts assertion and uses the unique identifier to look up or create a local identity

There are three key events in this sequence: the SP issues an authentication request to the IDP, a SAML assertion is issued, and a local account is looked up (or created). Additionally, the IDP-SP relationships established, and the names of these entities is relevant information.

An authentication request is a SAML message in which the SP details how the IDP should issue an assertion. A SAML assertion is a packet of security information including identity attributes and statements about how a user was authenticated. Looking up a local account requires that the user accounts on the IDP and the SP are, or have been, linked together. Users need to be able to find out which attributes are shared with which service providers, and should ideally be able to participate in making decisions about this.

4.1 Authentication Requests

When the user accesses a service provider site, which is a common way to initiate a federated authentication, the service provider sends an authentication request to the identity provider. The authentication request contains no personal information about the user, but does optionally provide some information about how the user identifier will be used. Of particular note is the name-id format which indicates whether a permanent link between a local account and the IDP user account will be used; known as *persistent* and *transient* identity federation. Other types of NameID exist, and are discussed is section 4.2. It is possible for both SP and IDP to support several NameID formats, one of which is chosen

at runtime; the possible values can be found by interrogating the *MetaData* [1] configuration information for both parties.

There is also an "AllowCreate" parameter that indicates whether the service provider wants to be able to create a local user profile based on the information passed later from the IDP. The intentions of the service provider can be relevant to users seeking to limit the number of personal accounts.

Thus, the two pieces of information judged relevant for end-users are whether transient or perisitent federation can be used, and whether the creation of local accounts on the service provider is allowed.

4.2 Assertions

Assertions, or more formally Security Asserions defined by the SAML v2 specifications, are where personal user data is exchanged. The assertion carries both the user identifier, *NameID* in SAMLv2 terminology, and user attributes. This information may be stored in a user profile on the IDP, dynamic information garnered from the authentication process, or third party information. Each assertion is addressed to an audience; a set of service providers.

Whilst the intended audience for an assertion can be of interest, it is very difficult to present this information in an understandable way to the user. There is no guarantee that the service provider will not pass on the information contained within the assertion, even if the actual assertion is only addressed to them. Audience is therefore judged not to be suitable for presentation to end-users.

The SessionIndex attribute to the AuthnStatement element can, in a privacy-wise poor implementation of the standard, contain user IDs, but this would be against the recommendation in the standard to use random impersonal values for the session index. Transfer of the session index is therefore not considered to be of interest to the user.

Likewise, the *NameID* value should be a random value for both persistent and transient formats, although other possible values are defined by the SAML v2 standards, including the use of email address, kerberos identifier, or x509 certificate distinguished name. These alternate, readable formats can be used if supported by both the IDP and the SP. This information is potentially of interest to the user, although rather than presenting the supported NameID formats, it is deemed easier to show the user the identifiers that are in use for each supported format. Note that transient identifiers are generated each time a session is created, so the values are indeterminate outside of a session.

Of prime importance to the user are the identity attributes; Attribute elements. Which attributes are exchanged can be reduced to a subset at runtime through the use of an *AttributeQuery*, but return by default all available attributes based on previous agreement. For example, in the OpenSSO SAML 2.0 implementation, the attributes available from the IDP and the attributes wanted by the SP are specified in extended metadata.

Thus, the information judged relevant to the end-users is the set of actual NameID identifiers that the IDP is prepared to use to identify the user to an SP, and critically the set of attributes that *may* be passed to the SP.

4.3 IDP-SP relationships

Within a *circle of trust* relationships exist between each IDP and SP. These relationships are expressed through the use of MetaData. In terms of SAMLv2, MetaData is a concrete specification in the form of an xml schema. This specification exists so that there is a standard way to exchange federation configuration information.

Most of the information in MetaData describes how to interact with the entity (an IDP or SP). Locations define the URL target endpoints, bindings describe the protocols and channels for interacting with these endpoints, and the enclosing tags associate the individual transactions with a higher level *profile*.

These locations are not of direct interest to the user, although the bindings and protocols could be. There are a number of choices in the SAMLv2 standards which affect the level of security, such as whether authentication requests are signed, assertions encrypted, or whether endpoints are protected by transport-level security (SSL). However, evaluating the security strength of a federated system is not deemed realistically achievable for laymen. There is no known standard third-party evaluation of security strength that could be presented to users either.

MetaData contains a list of NameID formats that are accepted by each entity. Since it has already been proposed that actual NameID identifiers should be shown to users, the list of NameID formats need not be displayed and explained to the user.

Thus it is only the IDP-SP relationships themselves that are of interest to the user. To be able to present which IDPs will assert a user's identity to which SPs, we need to be able to name each entity. This is detailed in the next section.

4.4 IDP/SP entity identifiers

Each IDP or SP has an entity ID as defined by the SAMLv2 specification, and which has no verified human-readable name. An entity ID may not be recognisable to a user, or may be difficult to separate from other entities. For example: in a real federation a trend emerged whereby the IDP name was also the name of the federation project, and so many service providers used this as part of their SP entity ID. Ie (anonymized):

Table 3. Example entity IDs

IDP	bigidp.sample.org
SP 1	bigidp.another.org
SP 2	company-bigidp.company.com

Many of the existing products that support SAMLv2 federation will use the hostname of a server as the entity ID. This can lead to cryptic entity IDs

such as "srv002-prod.brdg.company.com", which makes it harder for the user to understand which party is involved in the federation.

The accurate and secure translation of entity IDs is essential to providing users with the information and control they require. To address this issue, *nicknames* as defined by the *Petname System* [8] is proposed to give users a meaningful recognition of each entity involved.

The Petname System describes the process of naming people and entities in terms of security, memorability, and globalness. The system proposes the use of up to three names: a *key* which is global and secure, a *nickname* which is global and memorable, and a *petname* which is memorable and secure since each individual user sets their own petnames.

Since an entity ID accurately names an IDP or SP within a federation without being particularly memorable or understandable, this is a key. Rather than displaying to the user that "srv002-prod.brdg.company.com" is sending information to "idp.pqa.no" a more globally understandable name such as "The Norwegian Government" or "FlyMeFast Airlines IDP" should be used - this would be a nickname. Finally, a petname *could* be set in the dashboard by individual users to help them distinguish between entities with similar names; e.g. "My favourite airline". However, the main gain is perceived to be through the use of nicknames.

The SAML v2.0 specifications do not stipulate or facilitate the use of any identifier beyond the entity ID. Introducing a nickname in the Identity Dashboard would require gathering additional information about the identity and service providers and building up a local mapping of entity IDs to nicknames. However, nicknames would be equally useful within the federation for displaying information about which service provider the IDP is sending the user on to, for example. For this reason, the preferred solution is for each IDP to maintain a mapping to readable nicknames, and that these nicknames are communicated to the Identity Dashboard along with the information detailed above.

5 The Identity Dashboard

Having looked at identity federations and which information an end-user should ideally have access to, requirements can be stated for a proposed solution; a new component dubbed the *Identity Dashboard*. The Identity Dashboard is envisaged as a component that can be implemented alongside one or more network-based federations, and which would communicate with each IDP to collate information, and then present a web-based GUI to end-users where this information could be presented.

Figure 5 illustrates how the Identity Dashboard fits together with existing actors. Communication with end-users is via a web-based dashboard interface, whilst communication with IDPs would have to be standardized. In a SAML v2 context, then an extension or revision of the existing standards could be one approach. Alternately, a new standard with no dependencies on specific federation protocols could be created.

Based on the discussions above, there is a need for a solution which:

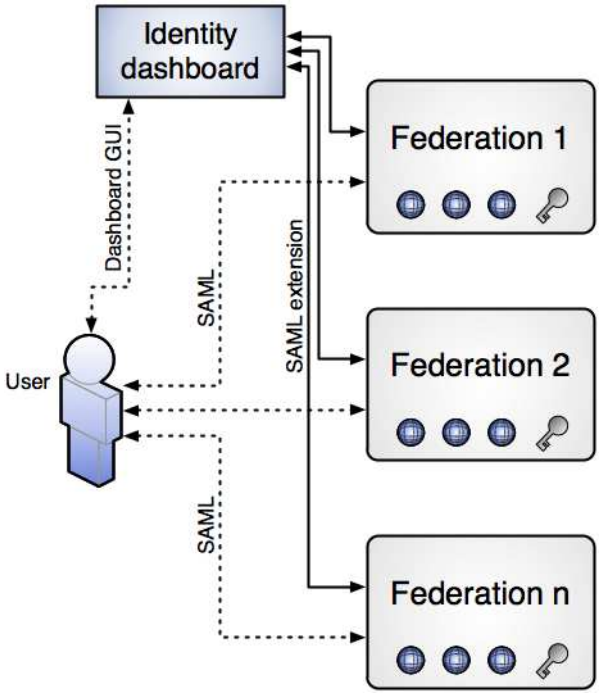

Fig. 5. Role of the Identity Dashboard

- Can collate information of interest for users
- Presents this information to users
- Provides a centralized information channel, both within and across federations, regardless of the federation model being used
- Can be trusted to give the user a better picture of their federated identity

The location of such a solution affects the requirements; a local dashboard application on the user's client machine would involve direct network contact between the client and each identity provider, client platform security would be a major factor, and mobility would have to be considered. A remotely hosted dashboard would require a 3rd party interested in providing such a service, and could potentially present a privacy threat if user identifiers are stored and/or used for other purposes by the dashboard. In this paper the Identity Dashboard is suggested as a common web-based component, although both approaches have their merits.

A proposed solution for handling these requirements is detailed in the following sections.

5.1 Collating information

The following pieces of information were highlighted as relevant for users:

1. Name of participating entities (translation of entity IDs) per federation
2. List of the NameID identifiers that the IDP is prepared to use to identify the user in each IDP-SP relationship
3. List of attributes available in each IDP-SP relationship
4. Details of whether persistent or transient federation is supported and/or used for each IDP
5. Whether accounts will be created on-the-fly when a user performs a federated authentication without previously having an account
6. Additional *soft* information about each federation (rules, legal frameworks, contact details)

The IDPs in each federation are the datasource for all this information. Thus a connection between the Identity Dashboard, and the involved IDPs should be sufficient to garner this information. Note that due to privacy and security reasons, it is suggested that the Identity Dashboard should not store this information unless essential; information should be provided at runtime from each IDP at the request of the user.

As mentioned above, standardising how this information is retrieved from the IDPs is essential to realising the Identity Dashboard. The standard would also have to address security - how will the IDP know to release information to the dashboard? There are several approaches which could be considered, including using the ID-WSF and ID-SIS specifications from Liberty to create a new standard identity service which would retrieve this information. In such a scenario, the user would have to authenticate to an IDP in order to retrieve the relevant information. This would be a natural approach from a security perspective, although not particularly user-friendly as this would require the user to log in multiple times to access information.

The soft information (rules, legal frameworks, and contact details) is the only information that is not systematically available to the IDP. It is however natural that this information be made easily available and that the store for this information would be the IDP.

A prerequisite for the above to be true is that the Identity Dashboard must be aware of which identity federations the user participates in. A possible approach is for IDPs to register with the dashboard, where the dashboard acts as a discovery service. The user could then be presented with a list of known IDPs/federations, and asked to indicate those that are appropriate.

5.2 Presenting information to users

When the Identity Dashboard has securely collated information from one or more IDPs where the user has an account, the information must be presented to the user in an understandable way. The channel for presenting information is

Fig. 6. Selecting federations where the user is active

Fig. 7. Relationships with the federation

a web-based dashboard accessed over HTTPS. Figure 6 sketch showing the first step: selecting identity federations that the user participates in.

Beyond this, the user would potentially be required to authenticate to the IDP involved, whilst the Identity Dashboard negotiates with the IDP using ID-WSF. This step would be as per a normal login to the identity federation. Following this, the user would be presented with a list of relationships this IDP has with other SPs, and in a proxy federation, other IDPs. A proxy federation is where an IDP is itself a service provider, and which trusts a different IDP outside of the original circle of trust. See figure 7.

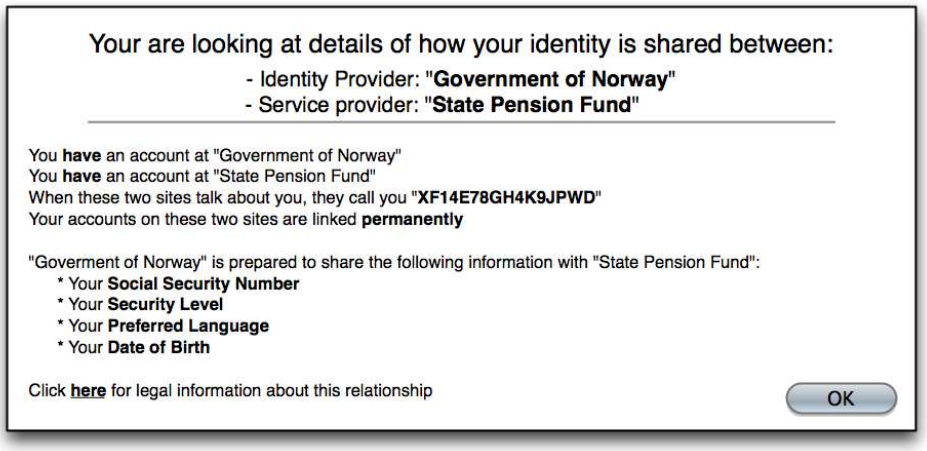

Fig. 8. Details of the IDP-SP relationship

When a user wishes to look at an account link in more detail then clicking on a relationship should present the value and nature of the NameID identifiers, which user attributes are shared, and whether an account will be created for the user automatically on first-login. An example of this is shown in figure 8.

A collated view showing selected details in a single table of federations and relationships would also be useful, although it would be expected to require multiple log-ins to the IDPs supplying information to the dashboard.

6 Conclusion

The Identity Dashboard is presented as a possible approach to enabling users to control their federated identity. Should such a system be implemented, then there are a number of details and alternatives to consider that have not been fully explored in the scope of this paper. There are also some outstanding challenges.

As discussed in section 5, there are arguments for staging the Identity Dashboard on a web-based server or on the client system. Should extended standards be developed which would describe passing the necessary information from the IDPs to an Identity Dashboard, then both client-based and server-based dashboards would likely use the same protocols. The location is therefore not limited by the stance taken in this paper.

An aspect which is not further explored in this paper is to use the Identity Dashboard as a channel for interacting with end-users. Potential scenarios include acting as a Liberty ID-WSF interaction service, and providing standard tools for users to unlink accounts. A particularly relevant case is that user interaction during account linking is not covered by any Oasis or Liberty standard at this point in time. The implementation and design of such interaction can be difficult, and is often oversimplified reducing the number of decisions the use can make. In the worst case, account linkage may occur without the user being informed of the consequences at all. Could the Identity Dashboard play a watchdog role in account linking? These ideas are not currently included in the scope of the Identity Dashboard, but may be relevant for a server-based dashboard.

Central to the idea of the Identity Dashboard is passing information securely from an IDP to the dashboard. Asking the user to authenticate to the IDP as part of this process is presented as a possible solution, though user unfriendly. An alternative here could include viewing the Identity Dashboard as a valid identity provider in itself, which the target IDP trusts for the sole purpose of providing meta-information. This would however give rise to a number of security issues including account linking, trust issues, and increased security requirements of the Identity Dashboard itself. The question of authentication requires further exploration in order to find an optimal solution.

References

1. OASIS: Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS (2005)
2. Microsoft/IBM: Web Services Federation Language Version 1.1. Microsoft/IBM (2006)
3. Liberty-Alliance: Liberty ID-FF Architecture Overview, version: 1.2-errata-v1.0. Liberty-Alliance (2003)
4. Shibboleth Project: Shibboleth Architecture Protocols and Profiles, Working Draft 05. Internet2/MACE (2004)
5. Squicciarini, A.C. and Czeskis, A. and Bhargav-Spantzel, A.: Privacy policies compliance across digital identity management systems. In: SPRINGL '08: Proceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS, pp. 72-81. ACM, New York (2008)
6. Gevers, S. and Verslype, V. and De Decker, B.: Enhancing privacy in identity management systems. In; WPES '07: Proceedings of the 2007 ACM workshop on Privacy in electronic society, pp. 60-63. ACM, New York (2007)
7. Jøsang, A. and AlZomai, M. and Suriadi, S.: Usability and Privacy in Identity Management Architectures. In: The Proceedings of the Australasian Information

Security Workshop (AISW), CRPIT Volume 68. Australian Computer Society, Inc. (2007)
8. Stiegler, M., http://www.skyhunter.com/marcs/petnames/IntroPetNames.html
9. Cranor, L.F. and Guduru, P. and Arjula, M.: User interfaces for privacy agents. ACM Trans. Comput.-Hum. Interact, vol 13, nr 2 pp. 135–178. ACM, New York (2006)
10. Ackerman, M.S. and Cranor, L.F. and Reagle, J.: Privacy in e-commerce: examining user scenarios and privacy preferences. In: EC '99: Proceedings of the 1st ACM conference on Electronic commerce, pp. 1-8. ACM, New York (1999)
11. Ahn, G-J. and Lam, J.: Managing privacy preferences for federated identity management. In: DIM '05: Proceedings of the 2005 workshop on Digital identity management, pp. 28-36. ACM, New York (2005)
12. Microsoft, http://msdn.microsoft.com/winfx/reference/infocard/default.aspx
13. Higgins, http://www.eclipse.org/higgins
14. Ahn, G-J. and Ko, M. and Shehab, M.: Privacy-enhanced User-Centric Identity Management. In: Proceedings of the IEEE International Conference on Communication IEEE (2009)
15. Suriadi, S. and Foo, E. and Jøsang, A.: A User-centric Federated Single Sign-on System. In: 2007 IFIP International Conference on Network and Parallel Computing. IFIP (2007)
16. Earp, J.B. and Antn, A.I. and Aiman-Smith, L. and Stufebeam, W.H.: Examining Internet Privacy Policies Within the Context of User Privacy Values. In: IEEE Transactions on Engineering Management. IEEE (2005)