B.F. Marques, T. Nebe, R.F. Oliveira (Eds.): Proceedings of the 12th Annual Workshop of HP OpenView University Association (HPOVUA 2005);
pp. 81–93, iPortalMais, Porto, Juli 2005

Federated Identity Management in Business-to-Business Outsourcing

Wolfgang Hommel
Leibniz Supercomputing Center
Barer Str. 21
Munich,
Germany
hommel@lrz.de

Helmut Reiser
Ludwig Maximilians University
Oettingenstr. 67
Munich,
Germany
reiser@nm.ifi.lmu.de

Abstract

While the outsourcing of IT services is a promising and cost-effective solution for many aspects of today's information and communication infrastructures, it poses new management challenges in the area of authentication, autorization and accounting (AAA). Due to the demand of cross-organizational AAA, traditional Identity & Access Management is presently developing into Federated Identity Management (FIM). However, existing FIM standardization efforts such as SAML still fail to bridge syntactic and semantic differences between cooperating organizations by requiring a common, federation-wide data schema. In this paper, we first demonstrate why this requirement is a severe obstacle for the efficient setup of identity federations by means of a real world B2B outsourcing scenario. We then specify an architecture, which extends SAML-enabled providers and solves the schema discrepancy issue based on XSLT transformations and a federation-wide schema correlation service. We also present its integration into the Shibboleth architecture, a popular open source FIM software.

Keywords

Federated Identity Management, SAML, Schema Coordination, Shibboleth

1. Introduction

The first generation of Identity & Access Management (I&AM) solutions focused on cost-effective and re-centralized internal management of an organization's employees, customers and business partners. From the organization's point of view, orchestrated authentication, authorization, accounting and auditing systems increase the intra-organizational security. The users also benefit, e.g. from single sign-on and web-based self services, which in turn reduces administrative overhead and service desk costs because the users can solve many problems themselves without requiring an administrator's manual intervention.

However, while those I&AM solutions cover the intra-organizational integration of services, they only work in relatively static environments with regard to the connected systems and applications. The next generation of I&AM, envisioned as

Adaptive Identity Management [14], thus strives for openness, increased flexibility and tighter integration into the underlying business processes.

Furthermore, in business-to-business (B2B) scenarios, inter-organizational identity data exchange is necessary, e.g. for Grid computing [6], Utility computing [15], supply chain management, and service outsourcing. For this purpose, several so-called Federated Identity Management (FIM) approaches have emerged, but they yet fall short of seamless integration into existing I&AM architectures [9].

One of the most serious problems, which the current FIM approaches have, is that they require a federation-wide common data schema for the identity information exchanged between the identity providers (e.g., outsourcing customers) and service providers. The demand for meta-directories and provisioning systems in I&AM scenarios already made it obvious that it is impossible to use the same data schema for each service; instead, the data must be pulled by or pushed to each service and application in the appropriate format. While most organizations do have a central identity management database, typically an LDAP-based enterprise directory, each organization's data scheme turns out to be quite different in practice, as we will show in section 2.

We show that current FIM approaches deal with this issue inappropriately in section 3, where also Shibboleth [5] is introduced, an advanced OpenSAML-based [1] open source FIM software, which is the de facto standard in higher education institutions. We also sum up a solution found for a similar problem in the area of federated database management systems (FDBMS). In section 4 we demonstrate how we extended the FDBMS solution to solve the common schema issue in FIM scenarios. We introduce an architecture for SAML-based providers which uses a federation schema correlation service and specify the related workflows. Finally, we describe how our components can be integrated into existing Shibboleth-based identity federations in section 5.

2. B2B outsourcing scenario

We present the following scenario to demonstrate which problems have to be faced when I&AM solutions are implemented independent of each other in several organizations and then are expected to interoperate. Thus, rather than going into the details about the services, which are to be outsourced, we focus on how data schemas for almost identical purposes have developed very antitheticly. We then analyze how existing FIM solutions cope with this problem in section 3.

The Leibniz Supercomputing Center (LRZ) is both the computing center of the universities and colleges in the area of Munich and the supercomputing center for scientific users from Germany and European Grid projects. The total number of active users exceeds 100,000, not including the constantly growing number of alumni who still are entitled to use services such as lifelong e-mail forwarding and access to an alumni web portal.

In the 1990's, there was a trend to service decentralization, i.e. many departments and chairs of the universities were running their own mail, file and web servers. As a result, university members who wanted to use some services offered by the LRZ, such as modem and ISDN dial-up lines, had to sign up separately for those services, i.e. they had to maintain their personal data, such as contact information, both at their university and at the LRZ. This inherently has lead to inconsistencies when someone made a change in one system but forgot to also incorporate it in the other.

Meanwhile, recentralization projects have been started. For example, the Technical University of Munich (TUM) has undertaken a project to build an enterprise directory as base for campus-wide I&AM. However, different subsets of a university's member data are required, e.g. by the university administration, which focusses on exams and study progress, and the LRZ, which requires accounting information. Not surprisingly, the data schemes used by the university enterprise directory and by the LRZ's internal I&AM solution have very little in common. While this difference could be explained by the fact that one of the organizations is a university, whereas the other one is a computing center, even the data schemes used by the two Munich universities have serious differences, as other data sources and recipients have to be integrated.

To reduce administrative overhead and increase the comfort for staff and students, it is intended that the universities make the necessary identity information available to the LRZ, so accounts can be set up automatically, e.g. for newly enrolled students and new employees.

However, the LRZ's I&AM solution cannot use the universities' enterprise directories directly, as the data schemes are so different. Both the syntax and the semantics of user attributes are incompatible, as can be seen from the following examples:

- *Date and address syntax.* E.g., a person's birth date is stored as one attribute in one directory, but as three separate attributes (day, month, year) in another. In different scenarios, a provider might require the person's age, but not its exact birth data for privacy reasons. Although this information can be derived from the available data, it is not explicitely stored and thus cannot be requested directly. Similar problems are faced with postal addresses; e.g., one directory offers three arbitrarily fillable lines of text to store an address, while the other has separate attributes for name, street, postal code and city.
- *Study course identification.*
  - Syntactical issue: Study courses are stored as attributes of student objects in one university's directory but as separate objects in the other's due to very different data models.
  - Semantical issue: Study courses, which are offered by both universities often have slightly different names. Furthermore, for the LRZ, the exact name of the study course is not relevant, but its affiliation to a faculty is; e.g., "media informatics" students are assigned the more general role "computer science student" and the CS faculty is contacted in case of troubles such as account abuse.
- *Attribute semantics.* Values of attributes with identical names can have differ-

ent semantics. For example, a person's `nationality` attribute has a value of "German" in one directory, but "DE" in the other.

As as part of a complete FIM solution, one expects means to overcome these syntactical and semantical differences and thus a seamless integration into the I&AM solutions at both ends. We will analyze the state of the art next.

3. State of the art

When FIM technologies such as SAML [10] (an OASIS ratified standard), Liberty Alliance [12] (an industrial consortium founded by Sun) or WS-Federation [11] (by IBM and Microsoft) would be applied to the scenario introduced above, the universities would act as Identity Providers (IDPs) for their students and make the data which is required by a Service Provider (SP) available on demand through dedicated protocols. As the data only needs to be maintained at the IDP, its up-to-dateness and quality is much higher compared to isolated and independent I&AM solutions, which have to acquire and maintain the user data separately.

FIM enables the SP to request arbitrary information about the user from the IDP; the IDP looks up these *attributes* and returns them to the SP if permitted by individual privacy policies. Yet, requesting an attribute requires to know its name and how to interpret the response. In other words, the IDP and the SP must have a common understanding of the naming, syntax and semantics of the exchanged attributes to be interoperable. While this requirement sounds harmless in theory, it is a killer criterion in practice. As each of the parties involved in the scenario described above uses a different data scheme to manage its users internally, attribute requests made by the LRZ would not be understood by either of the universities in general. Finding a common schema for all parties is practically impossible due to the different intra-organizational requirements and the costs for changing existing systems. This is especially true when the number of participating organizations increases, i.e. the common schema approach – which is also known as Schema Integration in database management – suffers from bad scalability. Those three major FIM approaches currently currently have other deficiencies as well, as shown in [9].

Shibboleth [5] is an open-source FIM software, which is based on OpenSAML [1] and especially popular among higher education institutions [13]. As Shibboleth is SAML-based, it requires a common data schema for the whole federation; however, it provides better integration into local I&AM systems by allowing to define *attribute mappings*, which is a feature various commercial FIM products offer in the same way, too. Those attribute mappings allow to *rename* attributes; for example, if the attribute `birthdate` was requested, it could be looked up as `DateOfBirth` in the local database. However, an attribute's value still can neither be composed of several database fields nor modified on-the-fly.

Thus, for the scenario described above, none of the current FIM approaches nor Shibboleth, which offers attribute mappings, can be used out-of-the-box, i.e.

they demand that all participants use the same data schema, which would require massive modifications to the already existing local I&AM solutions.

Similar issues have been researched in the field of Federated Database Management Systems (FDBMS); it has been shown that the Schema Integration approach, i.e. finding a common schema for all involved databases, suffers from exponential complexity [16]. A promising solution to this problem is the Schema Coordination approach, which is similar to the attribute mapping described above, but instead of being local to each site, a central Attribute Correspondence Matrix (ACM) is maintained which holds database column mappings for all involved databases and converts requests appropriately.

While the Schema Coordination approach has been successfully implemented for FDBMSs, it still does not solve the problem with composed attributes and incompatible attribute values in FIM scenarios. Also, using a central service for requesting data would be a single point of failure and violates the Liberty Alliance's postulate for autarkic federation members and decentralized communication. In the next section, we describe how we extended this approach and applied it to SAML-based FIM scenarios.

4. Enhanced data schema correlation for SAML-based identity federations

Our goal is to enhance the interoperability of FIM solutions with existing I&AM systems by

- letting both the identity and the service provider work with their individual local data schemas internally, and
- minimizing the administrative overhead which is necessary to set up attribute mapping and conversion rules,

while still maintaining full compatibility with the existing FIM standards. We reach this goal by the following means which are described in more detail below:

- extending the Schema Coordination approach, which was described in section 3, by processes and tools to additionally convert the attributes' values into the required target formats, and
- introducing a (logically, not necessarily physically) central Federation Schema Correlation Service (FSCS) for identity federations, which basically is a common conversion rule repository; it does not do any payload data conversion itself, so each provider's autarchy is not affected.

The following specification is based on the SAML architecture, but could be applied to Liberty Alliance and WS-Federation setups as well. We have chosen SAML because the Liberty Alliance specifications are based on SAML and WS-Federation also supports SAML assertions; furthermore, Shibboleth is based on SAML, and our implementation's target is Shibboleth, as described in section 5.

We first describe how our components fit into the SAML architecture. Then, we specify the attribute conversion workflow in section 4.2 and the tasks of the FSCS in section 4.3.

4.1 Extended SAML architecture

Figure 1 on page 7 shows the standard SAML architecture, extended by an attribute converter component local to the identity provider and the federation's FSCS component. The attribute request workflow is as follows:

1. A new request for attributes is sent by the service provider to the identity provider by means of a standard SAML request.
2. Instead of directly looking the attributes up in the local database or enterprise directory, the request is forwarded to the identity provider's attribute converter component. This component can be integrated into the SAML policy decision point (PDP) or implemented as stand-alone web service which the SAML PDP uses.
3. The attribute converter knows both the requester's data format and the locally used data schema by mechanisms described below. Based on the conversion workflow, which is described in detail in the next section, it converts the request so it can be applied to the identity provider's local data format.
4. The requested attributes are looked up in the repository, which in our case is an LDAP based enterprise directory. As the request is made in the local data format, no changes to this repository and other applications which are using it are required.
5. The retrieved attributes are filtered using an Attribute Release Policy Decision Point (ARP PDP). ARPs can be used to restrict which attributes certain services providers can request and thus protect the user's privacy. While ARPs are not part of the SAML standard, the Liberty Alliance introduces the concept of ARP PDPs and Shibboleth supports ARPs as well [2]. ARP processing is done after looking up the attributes, as the clearance to release an attribute might depend on its current value.
6. The attributes, which are allowed to be sent back to the service provider, are returned to the attribute converter component. This time, it converts the data from the locally used schema back into the service provider's format as described below.
7. The converted attributes are returned to the identity provider's SAML component, as if they had been looked up directly.
8. The SAML component returns the attributes to the service provider which requested them by wrapping them into a standard SAML attribute assertion.

Clearly, countersinking the attribute conversion component into the provider's internal request processing workflow maintains full SAML compliance when communicating with other providers. It accomplishes full integration of FIM requests into the local I&AM system by converting incoming requests to the locally used data schema. The price for this seamless integration is the necessity to set up appropriate conversion rules; we will discuss technical aspects of the conversion and how the administrative overhead for managing those rules can be minimized next.

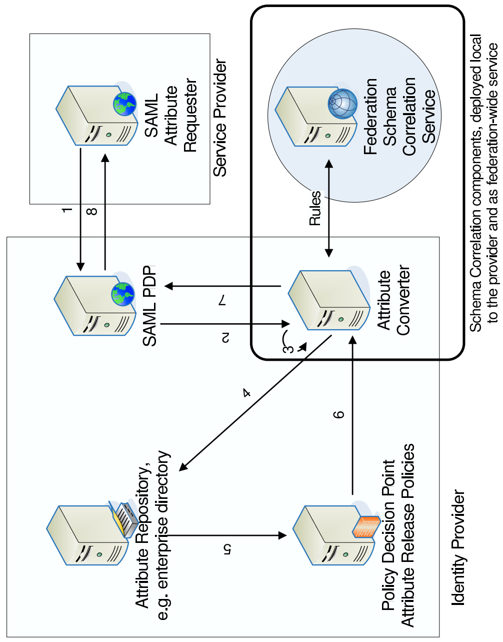

Figure 1 SAML architecture extended by Schema Correlation components

4.2 Attribute conversion workflow

The attribute converter's purpose is to transform both the incoming attribute requests into the locally used data format and the outgoing attributes into the original requester's format.

As an example, let us assume that a service provider has requested a person's `DOB` (date of birth) attribute and expects it in the format `yyyy-mm-dd`. However, the identity provider stores this information in three separate attributes, and also uses a 2-digit year format instead of a 4-digit one, as can be seen in figure 2.

Thus, whenever this service provider requests the `DOB`, the attribute converter must make sure that `bd_day`, `bd_month` and `bd_year` are looked up in the local identity repository. Analog, before the result can be returned to the service provider, the three distinct values for day, month and year of birth must be converted back into the `DOB` attribute in the format the service provider expects, as can also be seen in figure 2.

Conversion rules determine how this transformation is done. We are using XSLT stylesheets to formulate these rules, as XSLT [4] both allows to arbitrarily transform XML documents and offers a rich set of XPath [3] expressions and functions to specify sophisticated and complex transformation rules. Figure 3 demonstrates

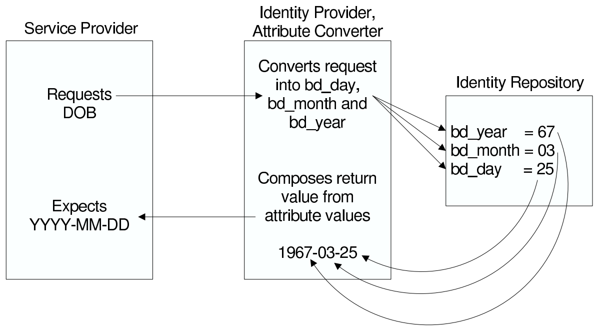

Figure 2 A simple schema mismatch example

the use of XSLT for the `DOB` example: First, the list of attributes to be looked up is converted to the local data schema, and then the results are combined into a response suitable for the service provider.

Besides for format conversions, XSLT can also be used to modify the attribute's content. Thus, not only syntactical, but also semantic modifications can be done. While it of course does require the understanding of one another's semantics, setting up such rules is in practice often eased by the fact that attribute values, such as a person's nationality, can only have discrete values. Thus, our extended

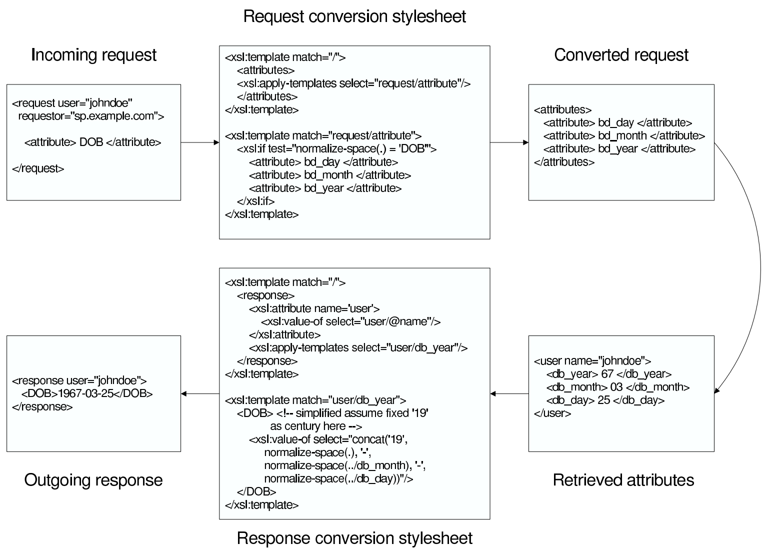

Figure 3 Example attribute conversion using XSLT

Schema Coordination approach does not only allow to map attribute *names* to each other, but also attribute *values*. Figure 4 demonstrates both types of conversion done in a single XSLT stylesheet.

Obviously, conversion rules only have to be specified for those communication partner pairs and attributes, which have syntactical or semantic differences. However, the number of rules required per site for larger identity federations with substantial data schemas would be $O(providers \cdot attributes)$. While the number of attributes has an upper boundary in the field of identity management for practical reasons, i.e. the information which needs to be exchanged about identities is limited, the overall complexity would still be $O(providers^2)$. In order to reduce the administrative overhead for setting up the conversion rules at each provider, we are thus introducing a central conversion rule repository, which exploits the fact that conversion rules between providers often are transitive. This Federation Schema Correlation Service is described next.

4.3 Federation Schema Correlation Service

The Federation Schema Correlation Service (FSCS) is a logically central conversion rule repository for all participants of a federation. Similar to other central components, e.g. discovery services, it can be offered by multiple federation members or third parties for high availibility reasons. As the required rules are also stored locally at each site and the FSCS only provides the rules, but does not do any payload data conversions itself, it is neither a single point of failure nor does it endanger the autarchy of each provider. The overall data maintenance thus remains distributed, as postulated e.g. by the Liberty Alliance.

The FSCS stores the federation's conversion rules in a matrix, similar to the Attribute Correspondence Matrix found in the FDBMS Schema Coordination approach (see section 3). Each matrix cell is identified by a tuple $(sender, recipient)$ and has the following content:

- A flag which indicates whether the sender or the recipient is doing any necessary conversions. There is no preference whether service providers or identity providers should do the conversion in general, as this heavily depends on the actual number of service providers and identity providers in the federation and which members support our extension at all – it is sufficient that either of both communication partners does.
- Two hash tables which hold the rules for converting the attribute requests and the responses. The keys in these hash tables are the names of the requested attributes. Each value in the hash tables is an array; each element in the array can be either a conversion rule, i.e. a pointer to an XSLT stylesheet, or a link to another cell in the matrix. Using such links, identical rules do not have to be stored multiple times. Both the XSLT stylesheets and the matrix cells are identified by URNs, i.e. each element holds an URN, for example `https://fscs.federation.example.com/matrix/sender/recipient/reqconv/1/rule.xsl`. If the value array has more than one element, multiple XSLT stylesheets will be applied in a pipeline. This adds a level of indirec-

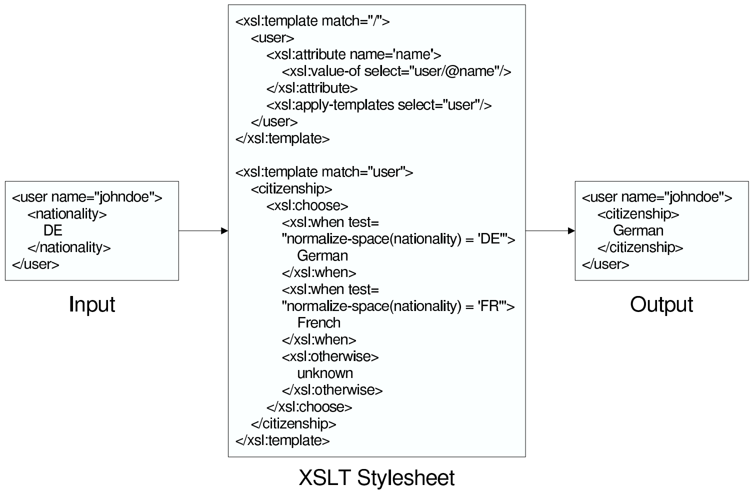

Figure 4 Example for mapping both attribute names and values in XSLT

tion, so existing XSLT code can be reused even more efficiently by allowing the combination of stylesheets.

- For rules which do not only change the syntax of an attribute, but also its values, a list of discrete values for which the rule can be used is provided as array.
- Meta-data, such as a timestamp when the cell's content was modified last time, which are required for the attribute converter side caching mechanisms.

Note that rules for both directions $a \rightarrow b$ and $b \rightarrow a$ need to be set up separately. This is necessary because XSLT stylesheets cannot be executed "backwards", which is obvious because the transformations can be lossy, e.g. if a person's name is transformed into its initials.

In practice, if rules are set up for $a \rightarrow b$ and $b \rightarrow c$, transitivity allows an out-of-the-box communication $a \rightarrow c$ if only the subset of attributes is used, for which conversion rules $a \rightarrow b$ as well as $b \rightarrow c$ exist. Thus, the administrative overhead to complete the ruleset for $a \rightarrow c$ communication is reduced to specifying those rules for which no $a \rightarrow x \rightarrow c$ exists, with $x \in M$, M being the set of federation members.

The FSCS provides methods for reading and writing those matrix cells, as well as for searching conversion rule paths $a \rightarrow b_1 \rightarrow \ldots \rightarrow b_i \rightarrow c$, with $i < |M|$. If $i \geq 1$, the resulting rule path can only be used, if it does not contain value-changing rules or the actual attribute value is supported by each such rule, which must be tested for the next rule in the pipeline after each transformation. For attribute values which fail one of these tests, dedicated rules $a \rightarrow c$ must be formulated.

Access to the FSCS is restricted to the federation members, and can be protected, e.g. by verifying their SSL/TLS certificates. As it is a web service, the FSCS itself can be registered at the federation's discovery services, so its address does not have to be configured at each site manually.

5. Implementation and integration into Shibboleth

Shibboleth's identity provider component, called Origin, has built-in data connectors to retrieve attributes from LDAP servers, relational databases, and text files. It is open source, implemented in Java, and also provides extension hooks which allow to implement other connectors. Both the name of the requester, the so-called provider-id, which is a tuple $(serviceprovider, service)$, and the requested attribute are available to those custom connectors and can be used to select the appropriate conversion rules.

We are using Xerces [8] as XML parser and Xalan [7] as XSLT processor; both can be utilized in Java through the standard JAXP API. As it is a custom Shibboleth data connector, our attribute converter is running on the same machine as the Shibboleth Identity Provider, and thus the method calls between Shibboleth and the attribute converter are local (steps 2 and 7 in the workflow described in section 4.2). The conversion rules are cached as text files locally and the attribute values are retrieved from LDAP-based enterprise directories by using the standard JNDI API. Work is still in progress regarding

- the implementation of the central FSCS, especially an efficient rule path finding algorithm.
- the integration of the FSCS into discovery service mechanisms, as well as rule synchronization mechanisms between multiple FSCSs per federation, which are required for a decentralized high availibility solution.
- the reuse of the existing Shibboleth data connectors, i.e. only converting requests and responses instead of implementing a complete custom data connector. This requires modifications to the Shibboleth source code, but existing Shibboleth configurations and non-LDAP repositories could be retained.
- the notification of local administrators about
  - missing conversion rules which cause attribute requests to fail.
  - changes to 3rd party conversion rules, so their correct function can be verified manually or automatically by executing test suites.

6. Summary and outlook

This paper presented a real-world business-to-business scenario in which Federated Identity Management technologies must interact with existing local Identity & Access Management systems. We have demonstrated that incompatible data schemas are yet dealt with insufficiently by both existing FIM standards and current FIM implementations. As a solution, we proposed a data conversion component which utilizes a federation-wide available conversion rule repository. We presented an extension to existing FIM architectures, which is fully SAML-compliant, and specified the XSLT-based conversion workflow and its interaction with our Federation Schema Correlation Service FSCS, which is an enhancement to the Schema Coordination approach known from federated database management systems. Finally, we introduced our implementation and its integration into Shibboleth, an advanced open source FIM software.

Our further research focusses on related deficiencies which we have identified in current FIM approaches [9]. These include the treatment of Attribute Release Policies on the identity provider side and the Attribute Acceptance Policies on the service provider side, as well as federation-wide security services which could be integrated into central components such as the FSCS.

Acknowledgment

The authors would like to thank the members of the Munich Network Management (MNM) Team for helpful discussions and valuable comments on earlier drafts of this paper. The MNM Team directed by Prof. Dr. Heinz-Gerd Hegering is a group of researchers at the Ludwig Maximilian University Munich, the Munich University of Technology, the University of the Federal Armed Forces Munich

and the Leibniz Supercomputing Center of the Bavarian Academy of Sciences. Its webserver is located at `http://www.mnm-team.org/`

References

[1] S. Cantor. OpenSAML 1.0 - an Open Source Security Assertion Markup Language implementation, 2004.

[2] S. Cantor. Shibboleth v1.2 Attribute Release Policies, 2004.

[3] James Clark. XML Path Language (XPath), Version 1.0. W3C Recommendation, `http://www.w3.org/TR/xpath/`, 1999.

[4] James Clark. XSL Transformations (XSLT), Version 1.0. W3C Recommendation, `http://www.w3.org/TR/xslt/`, 1999.

[5] M. Erdos and S. Cantor. Shibboleth-Architecture v05, 2002.

[6] I. Foster, C. Kesselman, and S. Tuecke. The Anatomy of the Grid — Enabling Scalalable Virtual Organizations. *Intl. Journal of High Performance Computing Applications*, pages 200–222, 2001.

[7] Apache Software Foundation. Xalan xslt processor. `http://xml.apache.org/xalan-j/`, 2005.

[8] Apache Software Foundation. Xerces xml parser for java. `http://xml.apache.org/xerces-j/`, 2005.

[9] W. Hommel and H. Reiser. Federated Identity Management: Shortcomings of existing standards. In *Proceedings of the 9th IFIP/IEEE International Symposium on Integrated Management (IM 2005)*, Nice, France, May 2005. In press.

[10] J. Hughes and E. Maler. Security Assertion Markup Language v1.1 Technical Overview, 2004.

[11] Ch. Kaler and A. Nadalin. WS-Federation specification, 2003.

[12] S. Landau and J. Hughes. A Brief Introduction to Liberty, 2002.

[13] Mikael Linden. Towards Cross-organisational User Administration. In *Proceedings of the 9th International Conference of European University Information Systems, Amsterdam 2003*, pages 140–147. EUNIS, 2003.

[14] M. Casassa Mont, P. Bramhall, and J. Pato. On Adaptive Identity Management: The Next Generation of Identity Management Technologies. Technical Report HPL-2003-149, HP Labs, 2003.

[15] Hewlett Packard. Management solutions across the utility computing continuum — strategic white paper, 2002.

[16] J. Leon Zhao. Schema coordination in federated database systems. In *Proceedings of the 4th Annual Workshop on Information Technologies and Systems (WITS94)*, 1994.