20th Bled eConference
eMergence:
Merging and Emerging Technologies, Processes, and Institutions
June 4 - 6, 2007; Bled, Slovenia

Enabling Attribute-based Access Control in Authentication and Authorisation Infrastructures

Christian Schläger[1], Torsten Priebe[2], Manuel Liewald[3], Günther Pernul[1]

[1]Department of Information Systems, University of Regensburg,
D-93040 Regensburg, Germany
christian.schlaeger / guenther.pernul@wiwi.uni-regensburg.de
[2]Cap Gemini Consulting Österreich AG, A-1020 Vienna, Austria
torsten.priebe@capgemini.com
[3]Accenture GmbH, D-61476 Kronberg im Taunus, Germany
manuel.liewald@accenture.com

Abstract
Attribute-based access control (ABAC) is a very powerful and flexible security technique making it possible to overcome limitations of traditional role-based and discretionary access controls. ABAC enables the dynamic handling of vast numbers of heterogeneous and changing resources and users, a task especially relevant for E-Commerce or distributed computing. With an authentication and authorisation infrastructure (AAI) in place, service providers could benefit from synergies and outsourcing possibilities and, simultaneously, strengthening their security level. In addition, AAIs could arbitrate between users' privacy issues and vendors' information demands, using privacy enhancing technologies. However, implementing ABAC is not trivial; nor is the derivation of attributes or metadata. This work proposes a solution to the demands for privacy aware, usable, secure, and outsourceable E-Commerce infrastructures with an AAI / ABAC combination. We introduce relevant technologies and an implementation that is evaluated. The prototype is based on the Liberty Alliance's ID-FF system, using XACML elements and classification tools.

Keywords: IT Security, e-business Models, e-business Architectures & Technologies, Authentication, Security Protocols, Access Control, Privacy

1 Introduction and motivation

In the changing environment of E-Commerce and distributed computing new demands on infrastructures and service providing have developed. For Service Providers (SPs) these demands include a higher level of security through fine grained access control and additional information about customers and the reputation as well as the possibility to outsource security services to 3rd party providers. Users require better usability with a Single Sign-On (SSO), central maintenance of account data, and the possibility to prove reputation and trustworthiness from one provider to another. Privacy protection has also become a major concern. Traditional techniques and methods can not satisfy these demands.
Of course, this list of user and provider demands is not complete. For an in depth discussion of E-Commerce stakeholder's demands see (Schläger et al. 2006) and (Schläger, Pernul 2005).

1.1 Service and security infrastructures for E-Commerce environments

At the dawn of E-Commerce only few providers offered services on the internet. Each vendor developed and maintained its own proprietary software solution as an isolated application. The main challenge then was to translate traditional brick-and-mortar business to the internet. Today E-Commerce has become ubiquitous. One user is related to many vendors and maintains numerous accounts and identities, each containing user profile data. The challenge has shifted from mere technical problems to convincing product offerings and diversified services in order to provide added value to customers in highly competitive markets. The Internet serves as a new platform for business transactions. Securing these transactions is crucial for E-Commerce providers (Katsikas et al. 2005).
Generally, we see contradictory requirements on E-Commerce servicing as depicted in Figure 1. A suitable infrastructure for modern E-Commerce providing needs to mediate between users and providers and offer sophisticated security and federation services.

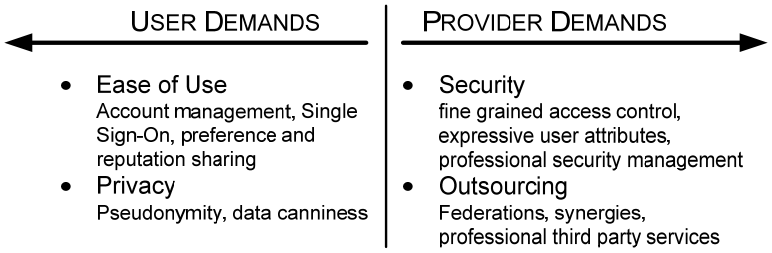

Figure 1: User and provider demands

E-commerce vendors can rely on infrastructures supporting their business processes. This can be extremely relevant for small and medium-sized companies not having the resources or the knowledge to secure their business or to maintain data storage facilities for sophisticated customer relationship management. In addition, these providers will especially benefit from synergies resulting from a larger customer and information base in a federation. Infrastructures could empower such a federation with complex, yet transparent to use, technology improving e-business models, processes, and security for E-Commerce providers.

1.2 Authentication and authorisation infrastructures (AAIs)

Known authorisation models for Access Control (AC) include discretionary (DAC), mandatory (MAC), and role based access control (RBAC). A discretionary model is not flexible enough for the changing portfolios and customer tailored resources or products offered in E-Commerce environments. The mandatory AC model is favourable for military purposes concentrating on information flow control making it inadequate for commercial purposes. Nowadays, RBAC is state-of-the-art for business products. However, Internet and E-Commerce environments lack a needed stable role structure. A permanent structure would be against the nature of the ever-changing Web.

Attribute-based access control (ABAC) is one of the latest developments in the field of authorisation and access control. With XACML – the eXtensible Access Control Markup Language (OASIS eXtensible Access Control Markup Language Technical Committee 2005) – an open standard has been proposed by the OASIS (Organization for the Advancement of Structured Information Standards) that is able to express policies based on classical access control models like role-based and discretionary access control (Priebe et al. 2004). XACML enables building complex policies that derive an access control decision from object and subject attributes. This standard facilitates providing dynamic, flexible, and fine-grained access control. An important standard for the exchange of security information between service and identity providers is SAML – the Security Assertion Markup Language (OASIS Security Services Technical Committee 2005) – also maintained by the OASIS group.

Service providers on the Internet are familiar with infrastructures providing basic security services. Authentication and Authorisation Infrastructures (AAIs) have started with basic Single Sign-On functionality and are nowadays able to manage the authorisation process and access control decisions. Two main architectures can be found: central ones like PAPI (Castro-Rojo, López 2001) or Microsoft's .NET Passport solution (Microsoft Inc. 2003) or federated ones like Liberty's Identity Federation Framework (Liberty ID-FF) (Cantor, Kemp 2005) or Internet2's Shibboleth (Cantor 2004).

AAIs help sharing security information about subjects and objects with other SPs or central services. Such information could be an assertion about the user's correct identification and authentication. Additionally, trusted sources can provide profile information. Using these attributes an access control model like ABAC (e.g., using XACML) can decide on the user's rights. Figure 2 shows the process of granting access to resources with the help of user, environment, and resource attributes.

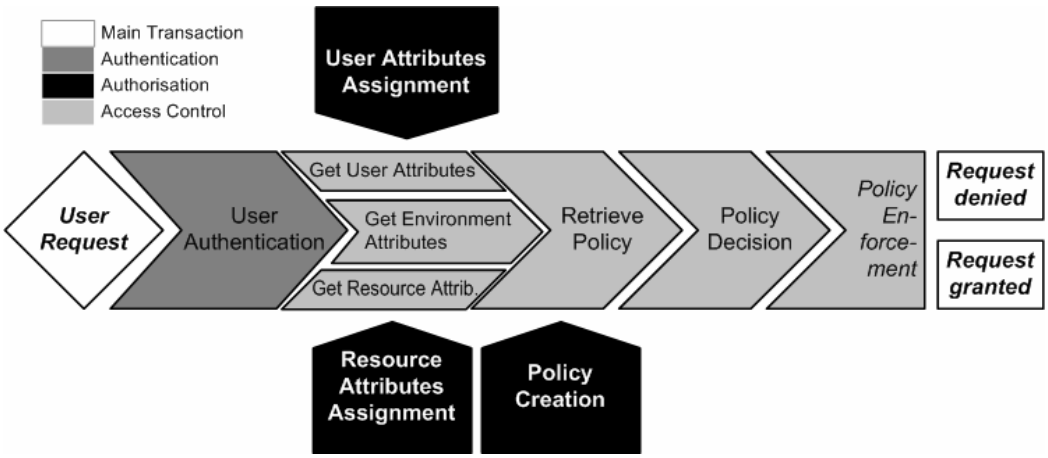

Figure 2: Attribute infrastructure security services

Attributes can contain identity and profile information. Despite this information's usefulness for access control, the user's privacy needs to be respected as well. To mediate between providers' wishes and users' demands trusted parties are needed, filtering or aggregating attributes. Aggregation or classification of attributes is a recognised privacy enhancing technology (PET) (Federrath 2005), additionally providing efficient ways to exchange data.

1.3 Enabling E-Commerce with attribute-based access control and AAIs

This work presents a combination of Authorisation and Authentication Infrastructures with attribute-based access control and privacy enhancing technologies. We suggest a new protocol and architecture to address the given problems. A prototype has been implemented, making it possible to evaluate the proposal.
Although AAIs are per se generic architectures, the term E-Commerce throughout the paper is used meaning business-to-consumer transactions (B2C E-Commerce). When talking about users a client e.g. a private customer is meant. The term Service Provider refers to vendors.

2 Related work and relevant technologies

AAIs make it possible to combine service outsourcing strategies with strengthened security. A special benefit lies in the accumulated user data over a federation: user profiles, buying patterns, and earned privileges. Identities can be transferred from one service provider to another making it possible to always use up-to-date address data or proof a good reputation acquired at one federation member. Comparative surveys on existing AAIs can be found in (Lopez et al. 2004) and (Schläger, Pernul 2005). Pfitzmann (Pfitzmann 2003) has analysed privacy issues in the Liberty ID-FF protocol and presented several enhancements. The idea of outsourcing non-functional tasks has been discussed in the field of software engineering (see e.g. Tanenbaum, Steen 2002).
(Katsikas et al. 2005) sum up requirements in providing secure E-Commerce. The shown need for flexible and dynamic access control in E-Commerce can be addressed with ABAC as presented in (Priebe et al. 2006) or (Yuan, Tong 2005). The basic idea is not to define permissions directly between subjects and objects, but instead to use their attributes as the basis for authorisations. For subjects, attributes can be static or dynamic. Age, current location or an acquired subscription for a digital library could be used as well.
Subjects and objects are both represented by a set of attributes and related attribute values. Permissions are defined between subject and object descriptors which consist of sets of attributes, conditions, and an operation that is to be executed. Environment attributes like time of day can be considered for the access control decision as well (Priebe et al. 2006).
Several techniques have already been mentioned in the cause of the introduction. To enable attribute-based access control the open standard XACML 2.0 (OASIS eXtensible Access Control Markup Language Technical Committee 2005) can be used. XACML has been applied with great success for implementations of ABAC. As XACML is agnostic to the exchange of attributes an adequate approach is needed. With SAML an open standard exists providing an XML-based protocol to transfer attributes (OASIS Security Services Technical Committee 2005). The integration of SAML with XACML has been proposed by (Anderson, Lockhart 2005).

2.1 SAML

SAML is an XML-based standard to describe security information which is communicated between system entities and domains. It defines the syntax of assertions about a subject and the processing semantics of these assertions. There are three different kinds of information which can be communicated with SAML: Authentication information, attribute information, and authorisation information. It can be used to provide requested assertions, authenticate subjects, and return the resulting assertion and perform a complete Single Sign-On and single logout process. In the assertion and the protocol definition are extension points declared to integrate additional functionality which is not covered by the SAML standard.

2.2 XACML

XACML is an XML-based standard to describe attribute-based authorisation rules and policies. Furthermore, it specifies rules to process and combine these authorisation rules and policies. XACML enables authorisation processes with fine granularity. XACML is based on rules which can be specified with the element `<Rule>`. A `<Rule>` has an attribute Effect. The value of Effect can be "`Permit`" or "`Deny`". Rules can be aggregated to policies and policies to policy sets.
A policy is defined with the element `<Policy>`. A rule-combining algorithm specifies how to evaluate several rules and how to treat different effects of rules in one policy. The resulting granularity and flexibility make XACML especially useful in changing, flexible, and heterogeneous environments.
A policy enforcement point (PEP) is the interface between resource and subject. It performs the access control for the resource. If the access requester (the subject) wants to access the resource the PEP forwards this request to the policy decision point. The request can contain additional attributes for subject, resource, kind of access and environment. The Policy Decision Point (PDP) computes the decision whether the subject gets access or not. For that purpose it requests additional attributes from the policy information point (PIP). The PIP retrieves these attributes and passes them back to the PDP. Now the PDP evaluates the policies and policy sets it got from the policy administration point (PAP) when it was initialised and sends the response to the PEP. The PEP permits or denies access for the subject to the resource depending on the response. Additionally it processes the actions stated in the `<obligation>` element of `<Rule>` or `<Policy>`. As the XACML standard does not consider distributed environments or scattered attribute bases, it needs adaptation for AAIs.

2.3 Resulting technical requirements for AAIs

Based on the mentioned SAML and XACML functionalities and entities an appropriate AAI needs to support four steps or sub-services. Figure 3 shows their interaction.

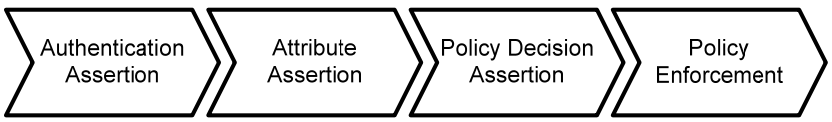

Figure 3: SAML/XACML services for ABAC in AAIs

In the easiest case the infrastructure provides only Single Sign-On. This would be done by giving out an assertion about correct authentication on a subject.

Combining this step with the next sub-service the AAI covers the transfer of attributes about users and resources. This could be implemented as interfaces which allow a resource to query a user's home domain about his or her attributes using SAML. Even more powerful are AAIs which can even come to a decision regarding a user's access request and then forward this decision to the resource. Finally, with the fourth and last sub-service, it is possible that the AAI enforces the decision by itself by installing a proxy system in between the client and the resource.

2.4 Lessons from existing AAIs

Various AAIs are available on the market, accessible as frameworks, products, or mere concepts. Among evaluated systems were Microsoft's .NET Passport, the Identity Federation and Web Service Frameworks from Liberty Alliance, Internet2's Shibboleth, the Spanish PAPI system, the privilege management infrastructures (PMIs) PERMIS and AKENTI, as well as Grid AAIs like CARDEA, CAS, GridShib, and VOMS. The detailed analysis can be found in (Lopez et al. 2004), (Schläger, Pernul 2005), and (Schläger et al. 2006). All solutions have a SSO functionality in common. However, neither of them is able to provide ABAC. At most, attribute exchange is supported. The implementation of SAML within Liberty's ID-FF and Shibboleth is exemplary for an open AAI. Grid systems and PMIs have developed means to compute an access control decision for the enquiring service. An enforcement of this AC decision is only realised by PAPI. PAPI resembles a proxy system covering the whole communication process between client and vendor. From an architectural point of view the adversaries .NET Passport and Liberty's ID-FF show the extremes of a centralised versus a federated infrastructure. When delegating security services to an AAI, the decision has to be taken for each module whether to centralise this service or provide it in a distributed manner. For more information on the consequences of allocation issues in AAIs and the resulting impact on functionalities and privacy see (Schläger, Ganslmayer 2007).

2.5 Privacy enhancing technologies

A common definition of privacy has been given by Alan Westin (Westin 1967): "Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others." When talking about privacy on the Internet, we usually mean informational privacy, which can be defined as "Self-determination of what information is known about a person and how it is used" (Schläger, Pernul 2005).
In (Federrath 2005) Privacy Enhancing Technologies (PET) are presented that try to guarantee confidentiality when using Internet-based communication. They can be defined as technologies minimising or avoiding personal data as well as safeguarding lawful processing of data. PETs aim at hiding the user's identity, making his actions unobservable to others and try to provide an unlinkability of user actions. They explicitly do not trust network operators or a single centralised station. From the privacy demands we have deducted three main functionalities for our ABAC enabled AAIs.

- First of all identity and privileges need to be separated.
- Secondly, data gained in the process of access needs to be stored in a distributed manner. Involved parties should use and gain as little information as possible.
- Finally, the user mustn't be forced to trust a predefined identity provider but should rather be able to choose among various providers the one of his liking.

3 Design goals for ABAC-enabled AAIs

This works proposed a new AAI concept integrating functionalities of ABAC based on open standards, mediating between user and provider demands in E-Commerce environments. We have shown that ABAC is preferable to other AC models for this scenario. With XACML we try to add ABAC functionalities to AAIs using an open and approved standard. Open issues with XACML for distributed usage have to be solved. The question of attribute information origin and privacy will be addressed with PETs. The following paragraphs summarise findings of the analysis of open standards, PETs, and existing AAIs and formulate design goals for the proposed architecture in section 4.

3.1 Elements of XACML and SAML

PEP – Policy Enforcement Point. Naturally, the PEP is closely connected with the SP. It is at this point where the final access control functionality takes place. The PEP could be realised as a proxy between user and vendor. However, the resulting bottleneck and the restrictions of a generic solution argue against a centralised approach. Furthermore, enforcing security decision outside the target application neglects inherent information about the application. For fine-grained access control the application's context must be incorporated.

PDP – Policy Decision Point. For the PDP the benefits of maintaining the decision process centrally have to be evaluated versus being a single-point-of-failure. We argue that for a given number of PEPs we need one PDP to guarantee performance and still use the benefits of maintaining it centrally. Following the arguments given in the introduction a local PDP is neither useful from a software engineering nor from an economical point of view.

PIP – Policy Information Point. The Policy Information Point gathers all relevant attributes for subjects, objects, and the environment. It is possible to assign a PDP one or more PIPs. The PIP communicates with the relevant attribute authorities collecting attributes that are then forwarded to the PDP. We see a need to integrate at least two PIPs into an access control decision. The PDP should make use of his own PIP collecting resource and environmental attributes. Another PIP is needed to perform the categorisation of user attributes. This needs to be done by a trusted authority. In our approach we have decided on the IdP to accumulate and categorise user data.

PAP – Policy Administration Point. In order to realise multi-layer policy architecture merging centrally stored high level policies with locally maintained low level policies the PAP needs to be able to derive policies from all service providers. The combined policies are loaded by the PDP at start-up. Centrally maintained PDPs should each have their own PAP.

SAML – SAML Assertions are responsible for the users' SSO. As an open standard it is predetermined to form the underlying communication technology for every open AAI approach. Its potential to sign, encrypt, and communicate any kind of attributes builds the basis for XACML decisions.

3.2 Attribute infrastructure

XACML distinguishes three classes of attributes: subject attributes (user), object attributes (resource), and environmental attributes. In our infrastructure the subject attributes are collected by the chosen IdP. Resource attributes and general environmental attributes are managed by the PDP's PIP. With this separation the SP is protected from revealing resource information to other members or competitors and the user can chose his most trusted IdP. The SP is supposed to use opaque and changing IDs for his products. The process of accumulating user attributes by the IdP is shown in Figure 4 using SAML/XACML

nomenclature. The IdP queries each SP if attributes about the identified user are available (step 1). If applicable the SP answers with a token (step 2).

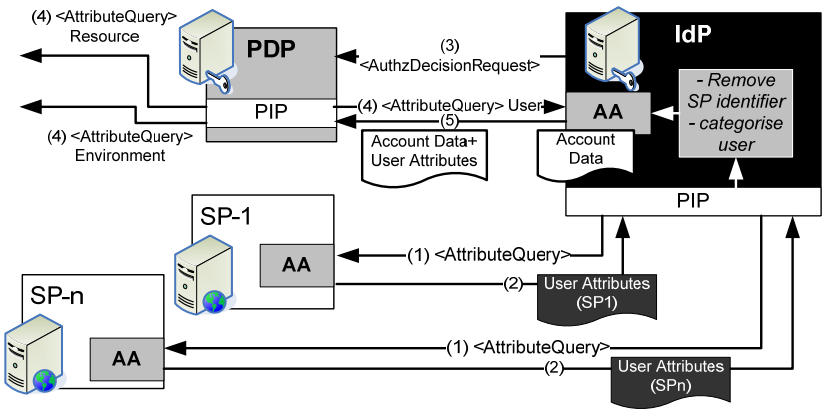

Figure 4: Collecting distributed attributes in a federation

The IdP uses the user's implicit profile information to complement the explicitly given account data. The request for a policy decision `<AuthzDecisionQuery>` (step 3) is send to the PDP which will request all three kinds of attributes from the IdP's PIP as well as from his own PIP managing resource and environmental attributes (steps marked with 4). After the decision is computed, it is send back to the requestor - the IdP (step 5).

3.3 Enhancing ABAC with attribute categorization

Data stored about the user can be classified in three levels, shown in Figure 5. A pseudonym is used as an identifier or Distinguished Name (DN). Adding to this pseudonym personal data like email address, true name, or shipping address a user can be associated with his true identity, regardless of the chosen DN. This data is given explicitly by the user and needs to be maintained. Even more information about the user is stored in his profile. The profile combines DN, explicitly given account data, and derived information from the users shopping patterns, payment history, or behaviour. This information can be called reputation. Please note, that in an open scenario, like E-Commerce, the notion of bad reputation is not feasible. Users behaving intentionally untrustworthy will not let this information be exchanged between SPs. They might simply apply for a new account.

In section 2.5 we have argued for distributed storage, data canniness, and user privacy. By nature, the idea of computing access control decisions based on attributes that contain user patterns and behaviour, reputation, and other personal profile data is in contrast with privacy issues. For our architecture we have implemented a distributed decision making process splitting and dividing information over different parties. Only in the unlikely case that the whole system and all members are compromised the users' complete profile information is accessible. Furthermore, user profile data is filtered.

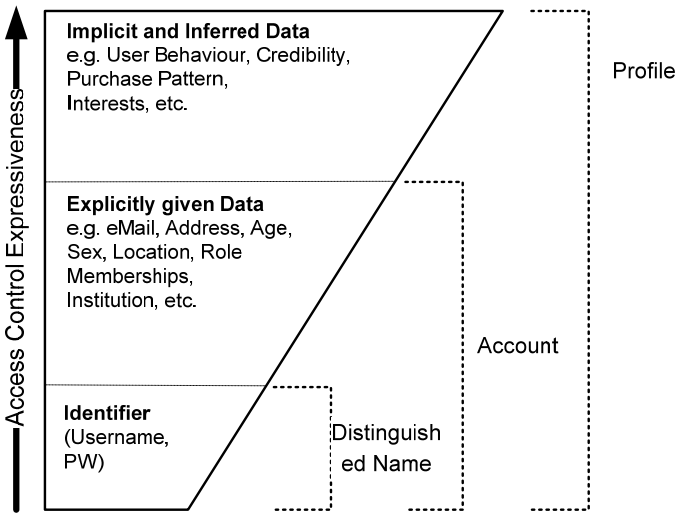

Figure 5: Attribute classes in AAIs and their Expressiveness

Although explicit user data is available through Customer Relationship Management (CRM) systems, it is sufficient to communicate relevant information only. A summary of different approaches to share trust and reputation in E-Commerce environments can be found in (Jøsang et al. 2005). For the prototype we adapted a simple trust management solution: We categorise business transactions into five different classes. We assume the members of a federation have agreed on general and binding processes to derive these classes from their CRM system. The categorisation is given in Table 1.
Exchanging user classifications rather then exhaustive buying patterns is also reasonable from a performance point of view. Furthermore, the classification of data enables SPs to define access policies much easier.

Table 1: Proposed user categories for reputation sharing

Category	Origin	Effect
Null	No data or insufficient data about the user, no categorization is possible.	
Standard	Data about the user is stored and was processed; no categorization for a special reputation or privileges could be made or the user opted against it.	The attribute issuer would grant no additional privileges.
Bronze	Recent business transactions have been completed satisfyingly over all.	The issuer would grant minor privileges.
Silver	Business transactions have been completed satisfyingly over a longer period of time.	The issuer would grant some privileges.
Gold	All recorded business transactions have been completed satisfyingly. The data stems from a longer period of time and from numerous transactions.	The issuer would grant full privileges.

With the proposed classification the PDP can evaluate the AC decision based on user attributes from his account (especially age, origin, and roles are important e.g. to comply with local legal regulations), his reputation status in the federation, environmental attributes from the accessed SP, and attributes describing the resource. Note that we have succeeded in separating subject and object identities from privileges to compute this decision. The PDP is not aware who accesses what.

4 Proposed protocol and implementation

For our prototype we made use of the main elements of Liberty ID-FF 1.1, namely distributed identity and service providers, and used SAML 1.0. The first step of the access control decision – the authentication – is handled as defined by the Liberty ID-FF protocol. In addition to these parties we introduce the XACML elements PIP, PAP, PDP, PEP, and a classification tool – the categoriser. As the XACML standard is very imprecise about the PIP we decided on using the PIP concept mainly as an interface on relevant databases and a as transformation tool of information and attributes into SAML. The originally proposed Context Handler has been substituted by this functionality. Figure 6 shows our ABAC-enabled AAI in a sequence diagram. The graph is in accordance with the UML 2.0 notation using SAML and XACML nomenclature. On top of the sequence we have included the information gained by the involved parties about the transaction in the process. Note that this process is generic. The requested resource can be any good, digital, or Grid computing service.

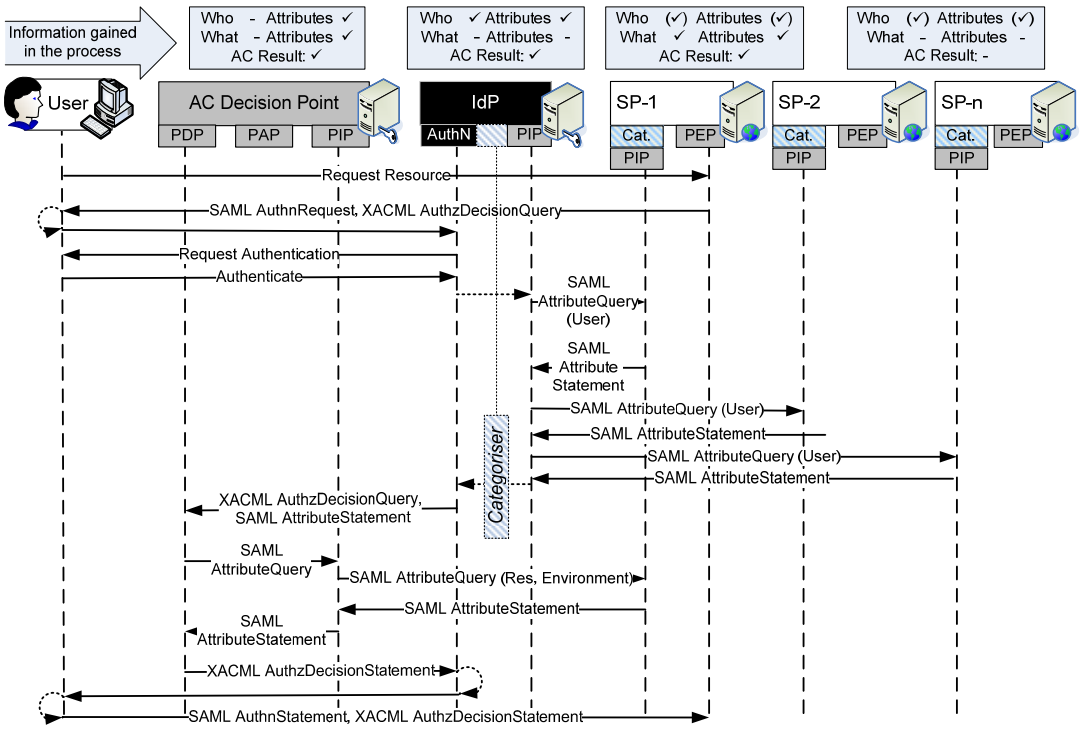

Figure 6: ABAC-enabled AAI prototype – sequence diagram

When the user tries to access a resource he is referred to his IdP. The IdP is derived either from a cookie stored in the user's browser or the user chooses from a list. This is the standard Liberty ID-FF SSO procedure. The SP (SP-1 in Figure 6) sends the IdP a SAML authentication request `<AuthnRequest>`. Additionally, he sends an XACML authorisation decision request, forwarding with it his identifier, a random, opaque user identifier, and a random, opaque resource identifier.

After the user is authenticated at his IdP the XACML component of the IdP – its PIP – collects all user profile attributes. Every SP in the federation is asked about user attributes. These attributes will be classified in one of the five groups according to Table 1. After the PIP has collected all attributes (respectively the

categorisations) their weighted value is calculated into one category. For our prototype this is just the average of all attributes. However, it is feasible to weight stronger classifications that have been derived over a longer period of time or consist of more business transactions. Adding account data to the classification the IdP can now send an AC decision request to the PDP with the identifier of the requested resource and all relevant user attributes. The PDP's PIP will collect resource and environment attributes. It is important that the SP's environment is used. For example for special offers with a specific deadline the time zone of the SP needs to be used. The time zone might be differing from the PDP's server. The AC decision is computed using the loaded policies. For our prototype simple policies have been generated to prove our concept. However, it is possible to use fine grained access control policies. The decision is send back to the requesting entity – the IdP. The SAML authentication statement and the XACML authorisation decision statement are referred back to the SP using the same communication channel as before. Although, Web Services could be used via the Liberty WSF we stay with Http Post requests at this stage. Such incremental developments will be applied in the next circle of enhancements. Finally, a local PEP at the SP will enforce the decision.

After the access request is terminated we find various data objects from this transaction at the various parties. Starting with the SP in question (SP-1) he knows what has been accessed in this transaction and the used attributes. He was asked by the IdP to provide the user classification. However, as in reality multiple requests will be made he should not be able to connect user identity and resource request. The only data he knows about the user is data already existing in his CRM. The brackets in Figure 6 symbolise this peculiarity. Naturally, the SP also knows about the outcome of the decision. Please note that, through the usage of opaque IDs, he gains no information about the user's identity via the SAML Authentication Assertion.

All other service providers know that a user has requested a resource in the federation. However, no further data is generated with the request.

The IdP knows who accesses a resource and his account data. Furthermore he gets informed about his classification or categorisation at the federation members and the outcome of the decision. We strongly recommend that a user can choose between various providers finding the one he trusts most due to the personal information aggregated at this point. Despite this recommendation our prototype right now features only one IdP. The policy decision point computes his decision only based upon the subject and object attributes. He is neither aware of the identity of the requestor nor of the product in question. Naturally he knows the decision.

For the implementation the SSO functionality of Liberty's ID-FF was used. For the XACML components we have started with SUN's XACML reference implementation changing and adopting where necessary. The classification is based on a simple Java-based computation. The classification will be developed to a rule-based service in the future. The existing implementation has also pointed to various issues of compatibility.

5 Conclusion

ABAC and AAIs are able to provide security services well suited for E-Commerce if combined logically and fostered on the appropriate technologies. The paper examined various architectural and technological models for AAIs. In addition to open standards, the proposed approach especially respects privacy demands. The given prototype is the consequence of the combination and adaptation of the open standards SAML and XACML with AAIs and privacy enhancing

technologies. We present – to our knowledge – for the first time a holistic solution for secure service providing with attribute-based access control in a service oriented infrastructure. Our protocol successfully mediates between provider and user demands. This is achieved through the integration of attribute infrastructures into AAIs to gain additional functionalities. With the introduced solution, service providers and especially small and midsized vendors can outsource services to the infrastructure and gain new functionalities for their business processes.
From a privacy perspective, we have succeeded in separating identity and privileges by introducing a categorisation mechanism that maps privacy-critical account data to reputation categories. The distribution of services and information leads to a minimum of required trust. The architecture is flexible enough to avoid the necessity of forced trust relationships for the user. For service providers, the proposed holistic approach to cover the entire security chain by implementing ABAC AAIs with PETs reduces threats and makes access control transparent for SPs. The usage of open standards and open formats avoids patchwork security for the federation.
The current proof-of-concept prototype will be enhanced to support SAML 2.0. With this update strong communication security based on a PKI will be introduced. Furthermore, the communication between servers will be enhanced to use Web Service standards.

References

Anderson, A., Lockhart, H. (2005): SAML 2.0 profile of XACML v2.0 (OASIS Standard), http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-saml-profile-spec-os.pdf.

Cantor, S. (2004): Shibboleth Architecture Protocols and Profiles Working Draft 05, http://shibboleth.internet2.edu/docs/draft-mace-shibboleth-arch-protocols-02.pdf.

Cantor, S., Kemp, J. (2005): Liberty ID-FF and WSF Protocols and Schema Specification, http://www.projectliberty.org/specs/draft-liberty-idff-protocols-schema-1.2-errata-v3.0.pdf.

Castro-Rojo, R., López, D. R. (2001): The PAPI system: point of access to providers of information., Computer Networks, Vol. 37, No. 6, pp. 703-710.

Federrath, H. (2005): Proc. of the International Conference on Trust, Privacy & Security in Digital Business (TrustBus '05), Lecture Notes in Computer Science (LNCS), Vol. 3592, "Privacy Enhanced Technologies: Methods, Markets, Misuse", Copenhagen, Denmark, 2005, Springer, pp. 1-9.

Jøsang, A., Keser, C., Dimitrakos, T. (2005): Proc. of the 3rd International Conference on Trust Management (iTrust 2005), Lecture Notes in Computer Science (LNCS), Vol. 3477, "Can We Manage Trust?" Paris, France, 2005, Springer, pp. 93-107.

Katsikas, S., Lopez, J., Pernul, G. (2005): Proc. of the 10th Panhellenic Conference on Informatics (PCI'2005), Volas, Greece, Lecture Notes in Computer Science (LNCS), Vol. 3746, "Trust, Privacy and Security in E-business: Requirements and Solutions", Volas, Greece, 2005, Springer, pp. 548-558.

Lopez, J., Oppliger, R., Pernul, G. (2004): Authentication and authorization infrastructures (AAIs): a comparative survey, Computers & Security, Vol. 23, No. 7, pp. 578-590.

Microsoft Inc. (2003): Microsoft Passport Review Guide, http://download.microsoft.com/download/a/f/4/af49b391-086e-4aa2-a84b-ef6d916b2f08/passport_reviewguide.doc.

OASIS eXtensible Access Control Markup Language Technical Committee (2005): eXtensible Access Control Markup Language (XACML), http://www.oasis-open.org/committees/xacml/.

OASIS Security Services Technical Committee (2005): Security Assertion Markup Language (SAML), http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security.

Pfitzmann, B. (2003): Proc. of the 3rd International Workshop on Privacy Enhancîng Technologies (PET 2003), Lecture Notes in Computer Science (LNCS), Vol. 2760, "Privacy in Enterprise Identity Federation", Dresden, Germany, 2003, Springer, pp. 189-204.

Priebe, T., Dobmeier, W., Kamprath., N. (2006): Proc. of the 1st International Conference on Availability, Reliability, and Security (ARES'06), "Supporting Attribute-based Access Control with Ontologies", Vienna, Austria, 2006, IEEE, pp. 465-472.

Priebe, T., Fernández, E. B., Mehlau, J. I., Pernul, G. (2004): Proc. of the Annual IFIP WG 11.3 Working Conference on Data and Application Security "A Pattern System for Access Control", Sitges, Spain, 2004, Kluwer, pp. 235-249.

Schläger, C., Ganslmayer, M. (2007): Proc. of the 2nd International Conference on Availability, Reliability and Security (ARES'07), "Effects of Architectural Decisions in Authentication and Authorisation Infrastructures", Vienna, Austria, 2007, IEEE.

Schläger, C., Pernul, G. (2005): Proc. of the International Conference on E-Commerce and Web Technologies (EC-Web'05), Copenhagen, Denmark, Lecture Notes in Computer Science (LNCS), Vol. 3590, "Authentication and Authorisation Infrastructures in b2c E-Commerce", Copenhagen, Denmark, 2005, Springer, pp. 306-315.

Schläger, C., Sojer, M., Muschall, B., Pernul, G. (2006): Proc. of the International Conference on E-Commerce and Web Technologies (EC-Web'06), Krakow, Poland, Lecture Notes in Computer Science (LNCS), Vol. 4082, "Attribute-Based Authentication and Authorisation Infrastructures for E-Commerce Providers", Krakow, Poland, 2006, Springer, pp. 132-141.

Tanenbaum, A. S., Steen, M. v. (2002): "Distributed systems: principles and paradigms", Prentice Hall, Upper Saddle River, N.J., USA.

Westin, A. F. (1967): "Privacy and freedom", Atheneum, New York, USA.

Yuan, E., Tong, J. (2005): Proc. of the IEEE International Conference on Web Services (ICWS'05), "Attributed Based Access Control (ABAC) for Web Services", Orlando, Fl, USA, 2005, IEEE, pp. 561-569.