Journal of Information Assurance and Security.
ISSN 1554-1010 Volume 8 (2013) pp. 107-118

Middleware Architecture for Cross-Border Identification and Authentication

Bernd Zwattendorfer
E-Government Innovation Center
Graz University of Technology
Graz, Austria
Bernd.Zwattendorfer@egiz.gv.at

Ivo Sumelong
OpenLimit SignCubes GmbH
Berlin, Germany
Ivo.Sumelong@openlimit.com

Herbert Leitold
Secure Information Technology
Center (A-SIT)
Graz, Austria
Herbert.Leitold@a-sit.at

***Abstract*— Many European states have issued electronic identities (eID) to its citizens since the early 2000s. Several have reached full coverage and usually high assurance credentials, such as smartcards, USB crypto tokens, or mobile phone eIDs are used. This lead to an impressive security infrastructure to authenticate at online services that, however, evolved as national silos – interoperability was no priority for a while. To overcome this, 18 European states have joined forces in the large scale pilot STORK. A SAML-based technical solution for cross-border eID federation between states has been designed, implemented, and finally piloted in a number of production services. In this paper we present the STORK middleware architecture that has been developed by Austria and Germany. Its main characteristic is a decentralized deployment that gives some end-to-end security and privacy advantages, but also needs particular attention to meet scalability challenges. This is compared to the STORK proxy model, an alternative centralized deployment approach that was chosen by other states. Federation between the two architectures is described, with particular attention to security and privacy aspects.**

Keywords- eID, electronic identity, middleware, STORK, interoperability

I. INTRODUCTION

The European Commission has recently (June 2012) published the proposal for an EU regulation on Internal Market electronic identification and trust services [1]. Pending upcoming political discussion in the Parliament and the Council to advance the proposal to a legal act, it is expected to establish a missing link on pan-European eID federation. This missing link is a legal basis for mutual recognition. The political will to advance to cross-border recognized eID dates back to the Manchester Ministerial declaration 2005. It states that by 2010 *"European citizens and businesses shall be able to benefit from secure means of electronic identification that maximise user convenience while respecting data protection regulations. Such means shall be made available under the responsibility of the Member States but recognised across the EU"* [2].

While the target date 2010 has been missed a bit, the political declaration enabled concrete actions. It has been further reinforced in the Malmö Ministerial declaration [3]. In fact, provisions for secure cross-border electronic services for EU citizens and businesses have already emerged from just a "desire" to a "must": An example is the EU Services Directive [4] that – in the services sector – grants a right for filing applications electronically from abroad. Such a right advances eGovernment from a voluntary provision to a public authorities' obligation. This also has implications on information security – think of how to uniquely and securely identify and authenticate unknown foreign citizens or businesses upon their first application.

The political decision is however facing technical and organizational reality. It in fact already has reached a pretty complex and heterogeneous state when the Manchester declaration [2] has been filed: Early adopters like Austria, Belgium, Estonia, Finland, or Italy started issuing smart cards to each citizen around 2003 and reached full penetration around 2005. Not just other countries followed like Germany, Lithuania, Portugal, or Spain, but also further credential technologies got widely deployed like mobile phone based eID, software certificates, or one time passwords. The European market of "higher assurance" credentials alone that can be used in an open environment, i.e. beyond a single application or sector sphere, amounts to almost 100 issuers of qualified certificates, bank ID that is popular in Scandinavian countries, etc. For an overview, see e.g. a European Commission study carried out for 32 states in 2009 [5].

To prepare the policy measures on mutual recognition and to get some hands-on experience with eID federation in such a heterogeneous environment, the European Commission together with fourteen European states – later extended to eighteen states – launched a large scale pilot "STORK" in 2008 (Secure idenTities acRoss boRders linKed1). The basic idea was to gain experience and to see in real production environments, where issues arise or one even might get stuck. Uncertainties that have driven the piloting idea have inter alia been trust framework considerations, security concerns, questions of accountability and liability, data protection or legal issues rooted in the procedural laws service providers need to meet. A governing principle of STORK was that the federation infrastructure shall not change existing national eID solutions, but shall be built as an interoperability layer on top of those. This recognizes the national responsibility of citizen identifications as a core sovereign act (cf. also the Manchester Ministerial declaration quote above).

One building block of STORK is the so-called "middleware model" (MW). It is an interoperability framework that has been developed by Austria and Germany,

[1] https://www.eid-stork.eu

as it best fits the user-centric eID infrastructure of those countries, but it is also appealing from an end-to-end security and privacy perspective. This paper discusses the middleware architecture. Therefore, the remainder of the paper is structured as follows: Section 2 settles the scene by dividing identity models into central, user-centric, and federated models following a classification by Palfrey and Gasser [6]. As federating eID is certainly no idea pioneered by STORK alone, but has been researched for a while, we sketch the most prominent technical solutions and frameworks in the related work section 3. We start discussing STORK in section 4 where the basic framework is described and the two interoperability models are explained. These models are referred to as "STORK middleware" (MW) and "STORK Pan-European Proxy Service" (PEPS). The core part of the paper is section 5 and section 6 which gets into the details of both the architecture and the implementation of the MW model. Security and privacy measures are discussed. Finally, we give lessons learned and draw conclusions.

II. Identity Models

Identification and authentication are by far no new issues, thus several different identity management systems have evolved [7]. In most identity management systems, user identification and authentication at a service provider is carried out via a so-called identity provider. Such an identity provider is responsible for user authentication and transferring user's identity and authentication data to the requesting service provider. Not all systems follow the same methodological approach; hence various identity models have emerged. For instance, some systems store identity data centrally, whereas other systems follow a federated approach. In this section we briefly describe three types of identity models (central, user- centric, and federated approach) based on the work of Palfrey and Gasser [6]. Distinction criteria are the storage location of identity data (i.e., central database, smart card, distributed storage). This classification of identity models can also be found in [8] and [9]. Other classifications like by Alpár, Hoepman, and Siljee in [10] distinguish between network-based and claim-based identity management models.

A. Central Approach

In the central identity model user and identity data are stored in a central database at the service provider or the identity provider. At first use of a service of a service provider, the user usually has to register at the service provider – when the service provider also acts as identity provider – or at an affiliated identity provider. Once registered, these identity data are managed and stored in central repositories in the service provider's or the identity provider's domain. When accessing a certain service or application at a service provider, the user must have been successfully authenticated at the identity provider before. After that, the identity provider forwards the identity data to the service provider. In this approach the user has no control anymore on which data are stored or actually transmitted to the identity information requesting service provider.

B. User-Centric Approach

In the user-centric model, the user herself always remains the owner of her identity data. Identity data are managed and stored within the user's domain (e.g., on a smart card) and are transferred to a service provider only if the user explicitly gives her consent. Using this approach a direct communication channel between the user and the service provider can be achieved and end-to-end security not involving third parties can be guaranteed.

C. Federated Approach

In this model user or identity data are distributed across various identity providers which have a trust relationship amongst each other. Such trust relationships are usually established on organizational level whereas enforcement is carried out on technical level. Commonly, the data repositories of the individual identity providers are linked and data can be easily exchanged. In most cases, data exchange takes place based on an agreement of a common identifier for a certain user.

Each of the three models has its specific characteristics. One may have advantages on privacy and user control, another on scalability. In fact, several representatives of each approach can be identified. We discuss a few in the next section.

III. Related Work

Numerous identity management initiatives and systems exist. We briefly introduce a couple of systems that gained importance either due its broad use, or as they established relevant standards.

First systems rooted from the need to manage employees' accounts and services in a single organization. User or employee data was simply stored in directories like LDAP (Lightweight Directory Access Protocol). In such a scenario, the scope of the identity management system was only focused on the staff of this single organization.

Since borders between countries or organizations are more and more decreasing (especially in the digital world), interoperable identity management gains importance. Identities must be managed and organized across multiple organizations or countries and hence identification and authentication data must be exchanged across domains. This means that identities must not only be handled within one definite context but must also be dynamic and changeable in different and more complex situations.

These challenges and requirements resulted in more sophisticated identity management solutions. Kerberos [11] for example was one of the earliest systems allowing secure and uniform authentication in unsecure TCP/IP networks. Additionally, Kerberos supports single sign-on (SSO), the ability of accessing several protected services in a distributed network by authenticating only once.

Due to the increasing popularity of the WWW the need for secure identity management systems arose also on application level. One such system supporting central authentication and single sign-on for several services on the

Web was Microsoft Passport2 (latterly called Windows Live ID). This system is seen as an example for a central identity model.

Other identity management systems came up such as the Liberty Alliance Project3 (that evolved to the Kantara initiative 4) or Shibboleth 5 . Both projects follow a decentralized architecture and allow SSO based on identity federation. Whereas the Liberty Alliance Project focused on federating enterprises, Shibboleth targeted on inter-connecting universities. The central authentication service6 (CAS) developed by Yale University is a further example of a federated network of universities for secure exchange of knowledge and technologies. As an open source system, CAS has been taken up by other sectors too.

Both projects, Liberty Alliance and Shibboleth, influenced the development of the current version of the Security Assertion Markup Language (SAML 2.0) [12]. SAML has been developed by OASIS and defines one of the most important standards dealing with SSO and identity federation. A similar framework constitutes WS-Federation [13], being part of the WS-Security [14] framework. Another decentralized authentication system defines OpenID7. OpenID is similar to the Liberty Alliance systems but uses URL-based identities for authentication.

Windows CardSpace8 can be seen as an example for a user-centric approach. A digital wallet installed on the user's client containing several so-called information cards, that reflect different identities, builds the core component of CardSpace. However, the development of its successor CardSpace 2 was abandoned in 2011.

Unique identification plays an important role for governments. The USA introduced its National Strategy for Trusted Identities in Cyberspace (NSTIC) [15] in 2011. It aims on the creation of a secure and trusted identity ecosystem facilitating access to public and private sector services.

Several countries – especially in Europe – have already rolled-out national eID solutions for eGovernment or eBusiness. Those eID solutions follow different approaches. The user-centric approach based on stronger authentication mechanisms is applied with secure tokens such as smart cards or mobile phones. Other approaches federate between authentication gateways, such BankID 9 in Sweden that piggybacks on internet banking authentication. Further approaches use central authentication gateways as identity provider, such as DigID10 in the Netherlands. Most national eID solutions rely on a Public Key Infrastructure (PKI) and the X.509 standard. The Modinis-IDM study [16], the IDABC eID country reports [5] or Siddhartha [17] give a comprehensive overview of national eID solutions in Europe.

IV. STORK Interoperability Models

This section gives an introduction to the STORK framework and its interoperability models. The aim of the STORK framework was to achieve cross-border eID interoperability agnostic from underlying technologies or infrastructures. Hence, the STORK framework takes already existing national eID solutions as a basis and builds an interoperability layer on top of it.

The STORK framework defines two models, the so-called PEPS model (Pan-European Proxy Service) and the MW model (Middleware):

The PEPS model is a proxy-based approach with identity intermediaries. A national gateway (the PEPS) serves as single interface to other countries and encapsulates specifics of the national eID infrastructure (i.e., the communication to service providers, identity providers, and/or attribute providers). The PEPS implements the protocols and functionality for cross-border authentication. In a cross-border authentication process, the PEPS is an intermediary between the service provider and the actual (foreign) identity provider. The PEPS asserts the service provider that a user has been successfully and properly authenticated by a foreign identity provider. The advantage of this proxy model is that each PEPS only needs to serve its national eID infrastructure and the common STORK protocol [18] for cross-border communication. Thus, in a cross-border scenario specifics of the national eID infrastructure are hidden from other involved entities of other countries. This also hides national or proprietary protocols from other countries, as the PEPS leverages to the common cross-border protocol.

In the MW model users directly authenticate at the service provider. This means that the service provider itself supports all desired identification and authentication methods. For supporting the middleware model, service providers install and deploy a so-called server-side middleware (VIDP – Virtual Identity Provider), which is operated in the service provider's infrastructure. This model can be associated to the user-centric models as classified in chapter II.B. It particularly preserves privacy because identity data are stored in the user's domain and no intermediaries are involved. Another advantage of this model is end-to-end security, as the user's eID (such as a smart card) can establish a direct communication channel to the service provider. A drawback is, however, that service providers need to integrate the various protocols and eIDs of

[2] http://www.passport.net

[3] http://www.projectliberty.org

[4] http://kantarainitiative.org

[5] http://shibboleth.net

[6] http://www.jasig.org/cas

[7] http://openid.net

[8] http://msdn.microsoft.com/en-us/library/aa480189.aspx

[9] http://www.bankid.com

[10] http://www.digid.nl

foreign countries. We will discuss in section VI how this has been met.

STORK implemented both models PEPS and MW, and its combinations. I.e., citizens from MW countries can authenticate at service providers of PEPS countries and vice versa.

Combining both models, four scenarios can be distinguished:

- A citizen from a PEPS country (PEPS infrastructure nationally deployed) wants to securely authenticate at a service provider in another PEPS country.
- A citizen from a MW country (MW infrastructure nationally rolled-out) wants to securely authenticate at a service provider in another MW country.
- A citizen from a PEPS country wants to securely authenticate at a service provider in a MW country.
- A citizen from a MW country wants to securely authenticate at a service provider in a PEPS country.

Figure 1 illustrates the first interoperability model (cross-border PEPS model) showing, on the one hand, the trust relationships between the participating entities and, on the other hand, the logical authentication process flow. A PEPS can either act as so-called S-PEPS (PEPS in the state of the service provider) or as C-PEPS (PEPS in the state of the citizen). An S-PEPS communicates with the service provider and the corresponding C-PEPS and thus depicts an intermediary between those two entities. In comparison, a C-PEPS receives authentication requests from an S-PEPS and triggers the identification and authentication process at an identity and/or attribute provider.

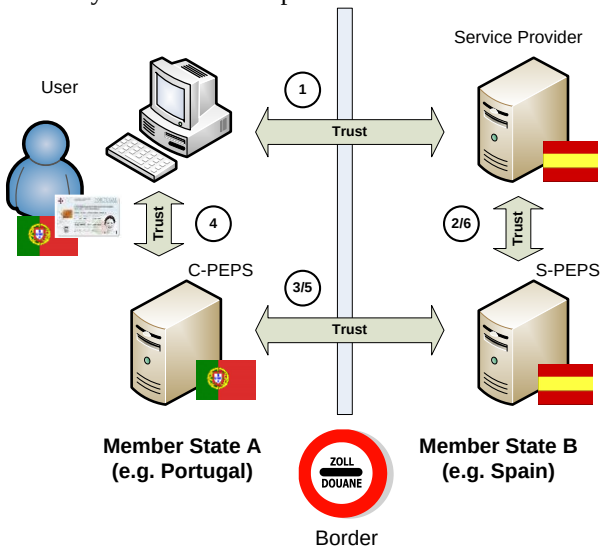

Figure 1. PEPS-PEPS Model

In this sample scenario a Portuguese citizen wants to authenticate at a Spanish service provider. It is assumed that the authentication process is started by accessing a resource at the service provider (Step 1) that requires authentication. The user is redirected to the S-PEPS of the service provider – the Spanish S-PEPS in our example (Step 2). At the S-PEPS, the user gets presented a country selection page. On this page, the user selects the country where she is originally from. This information is necessary to forward the authentication request and the user to her correct national C-PEPS (Step 3). The C-PEPS carries out the actual authentication of the user by contacting connected identity and/or attribute providers. For authentication, the citizen uses her national (Portuguese) eID (Step 4). Retrieved identification and authentication data is returned from the C-PEPS to the S-PEPS (Step 5). The S-PEPS in turn forwards these data to the authentication requesting service provider which now can grant or deny access to the protected resource (Step 6). If the authentication process was successful the Portuguese citizen has authenticated at a Spanish service provider using her own national Portuguese eID token.

During this authentication process, identity data is transferred or routed through several entities. The C-PEPS asserts the S-PEPS, and the S-PEPS asserts the service provider that the user has successfully authenticated. Because of this proxied architecture, a segmented trust relationship exists between the user and the service provider. Three point-to-point trust relationships are given: (1) between service provider and S-PEPS; (2) between the identity provider and the C-PEPS; and (3) between the C-PEPS and the S-PEPS. With the segmented trust relationships, the intermediaries must be secured properly. This is comparable to securing an identity provider's infrastructure. Note however, that a C-PEPS may proxy several national identity providers and an S-PEPS several service providers. This highlights the central, and thus security-critical role of a PEPS.

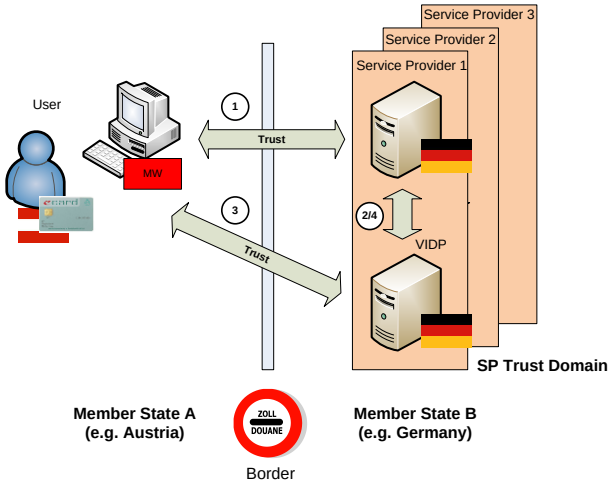

Figure 2. MW-MW Model

Figure 2 illustrates the STORK interoperability model where both countries rely on the MW approach. In the pure MW model no common national gateway exists. Instead, each service provider installs a server-side middleware

module (VIDP) directly in its domain. The VIDP is capable of several national eID token's security functions and manages the identification and authentication process for the service provider.

In this use case, an Austrian citizen wants to authenticate at a German service provider (Step 1). It is assumed that no security context between the service provider and the citizen has been established before and thus the authentication request is forwarded to the VIDP (Step 2). Based on the citizen's nationality, the VIDP triggers the corresponding national middleware module. In the STORK project as well as in the remainder of this paper the individual national middleware modules are called SPWare modules. For simplicity, the involved national middleware module (SPWare) is not shown. The SPWare module directly communicates with the citizen's eID token (Step 3). Received identity and authentication information is returned to the service provider via the VIDP (Step 4).

The foreign citizen is directly authenticated at the service provider via the VIDP and the corresponding SPWare. The VIDP (SPWare) communicates with the citizen's eID token without intermediaries. Both modules are installed and deployed in the service provider's domain, hence no explicit trust relationship between the service provider and the VIDP is required. The only clear trust relationship is given between the user and the service provider. As indicated in the figure by three planes, each service provider supporting the MW model operates a VIDP.

Figure 3 illustrates the authentication scenario where a user of a PEPS country (Portugal) wants to authenticate at a service provider located in a MW country (Germany). Basically, this scenario shows a combination of the PEPS and the MW model. In the first two steps on delegating the authentication to the VIDP, the authentication process flow is identical as in the pure MW model (Step 1 and 2 – cf. Figure 2). However, instead of triggering a national SPWare module the authentication request is forwarded to the C-PEPS of the user's home country (Step 3). The C-PEPS manages the actual authentication process (Step 4) and returns the identification and authentication data to the VIDP and the corresponding service provider (Step 5 and 6).

From the service provider's perspective the C-PEPS is an intermediary. The trust relationships thus are again segmented. This breaks the end-to-end security paradigm of the pure MW model. A trust relationship between the service provider's VIPD and the C-PEPS, and between the C-PEPS and the user is needed. Again, the VIDP and the service provider are in the same trust domain; hence no explicit trust relationship is necessary here.

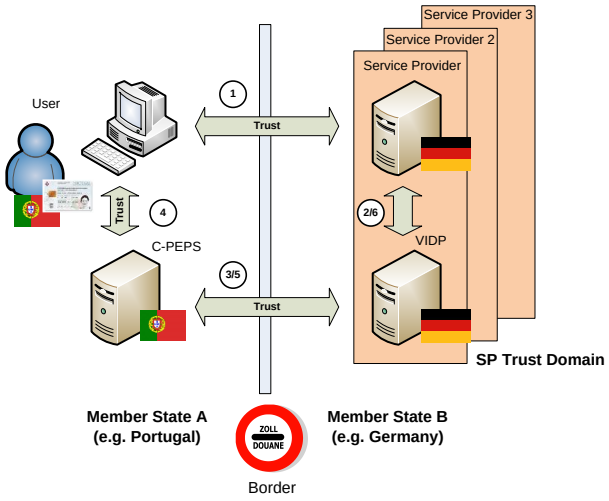

Figure 3. MW-PEPS Model

Figure 4 shows the final combination of the STORK basic models. In this scenario a user of a MW country (Austria) intends to authenticate at a service provider located in a PEPS country (Spain). Steps 1 and 2 on delegating the authentication to the S-EPS are as in the normal cross-border PEPS model scenario (cf. Figure 1). Step 3 is different because a VIDP, which is installed and deployed in the S-PEPS domain, is triggered instead of forwarding the authentication request to a C-PEPS. This VIDP manages the authentication with the citizen's eID token (Step 4). If authentication was successful the VIDP returns the authentication and identity information to the S-PEPS which forwards the data to the service provider (Step 5 and 6).

In this model the S-PEPS acts as a service provider in the classical MW model. The VIDP is hosted in the PEPS domain and hence no explicit trust relationship between those two entities is required. Similar to the PEPS-PEPS scenario segmented trust relationships exist between the service provider and the user and the service provider and the S-PEPS.

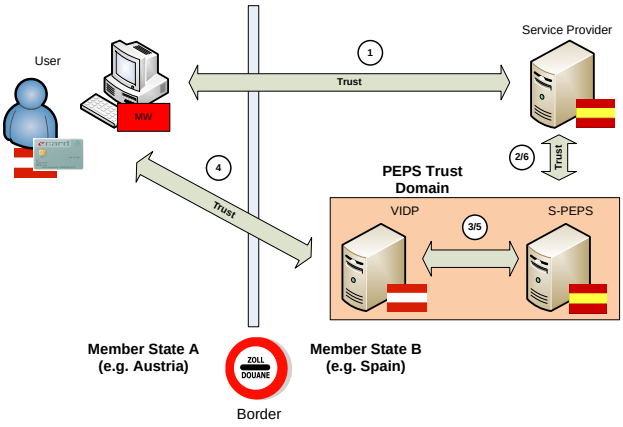

Figure 4. PEPS-MW Model

Aside authentication protocols, STORK defined quality authentication assurance (QAA) levels. It assigns each eID to one of four QAA classes. This is similar to levels of assurance (LOA) in other frameworks. QAA is not further discussed in this paper.

Summarizing, STORK is a framework that consists of two conceptual models, middleware and PEPS. Depending on which model countries of the citizen and the service provider opted for, four scenarios exist. In fact, common specifications and protocols have been designed so that STORK is seen as s single framework that supports both central and decentralized deployment. Identity and authentication data exchange is based on the well-known and standardized Security Assertion Markup Language (SAML) [12]. Details on the protocol for cross-border data exchange are given in the STORK interface specification [18].

V. Middleware Architecture

The middleware model represents the decentralized deployment option of STORK. It has merit from an end-to-end security and from a privacy perspective. It however faces the scalability challenge that service providers need to support several (possibly many) foreign eID tokens that can be based on different protocols. This asks for a modular and scalable architecture. This section describes the modular architecture of the VIDP, the main entity of the STORK middleware approach.

The MW model has been developed by Austria and Germany – both countries operating their national eID in a MW model: Austria has a national eID solution based on the MW concept and supporting several smartcards and mobile phone eID in use since 2003 (Austrian Citizen Card [19]). Germany has set up a MW infrastructure for the so-called "*neuer Personalausweis*" (nPA) [20] on national level in 2010.

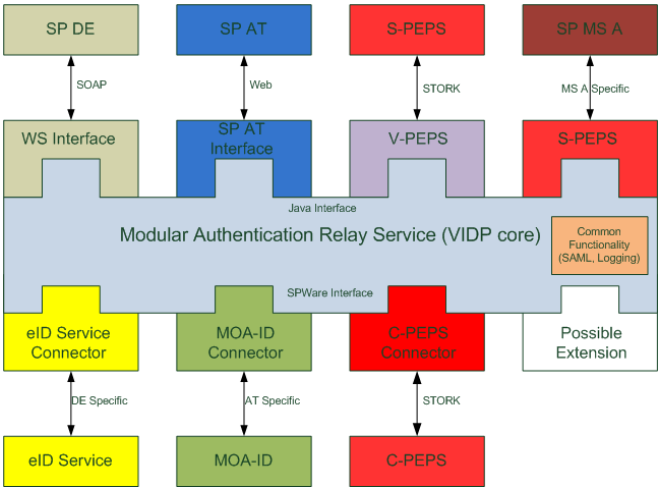

Figure 5. MW Architecture

Figure 5 illustrates the common MW architecture. To satisfy modularity and scalability requirements, it consists of a common component that can be extended by plug-ins and plug-ons for the national eID and SPWare protocols.

The common component is the Modular Authentication Relay Service (MARS). To integrate new countries' eIDs, two MARS-interfaces need to be implemented: (1) the Java Interface and (2) the SPWare Interface. Modules implementing the Java Interface handle incoming authentication requests of service providers (SP). These authentication requests are transformed and routed to the desired SPWare Connectors. The SPWare Connectors implement the SPWare interface and define connectors to the national MW module (SPWare). Figure 5 illustrates the SPWare Connectors to the German MW (eID Service) and the Austrian MW (MOA-ID).

Countries following the PEPS approach are also supported by this architecture. In this case (cf. Figure 3) the so-called C-PEPS Connector acts as SPWare Connector, which forwards an authentication request to the respective country PEPS (C-PEPS). Subsequently, the user authenticates at the according national PEPS which in turn wraps the identification and authentication data into a SAML token and returns it to the VIDP. The VIDP verifies the validity of this token and transmits the data through the respective national interface to the requesting service provider. The protocol for cross-border data exchange is based on the STORK interface specification [18], which is SAML.

The modular approach does not only provide the opportunity to easily integrate other countries' authentication systems but furthermore allows the conversion and restructuring of the VIDP to an entire PEPS. The realization by means of this architecture can simply be achieved by utilizing and invoking the modules S-PEPS and C-PEPS Connector together.

The implementation of this architecture contains the following components:

- *WS Interface*: This SOAP-based interface is used for authentication requests by German service providers. They send authentication requests to the VIDP and receive responses including identity and authentication data via this Web service interface.
- *SP AT Interface*: This interface is Web-based and supports authentication requests of Austrian service providers. They can use this interface for providing foreign eID access to legacy applications.
- *V-PEPS*: Via this interface the VIDP receives STORK authentication request messages from an S-PEPS. STORK authentication response message also pass this interface. In particular, this interface is involved in the cross-border PEPS-MW scenario (cf. Figure 4)
- *eID Service Connector*: This connector is responsible for the communication between the VIDP and the German eID service. The German

eID service constitutes the national German MW solution (SPWare).

- *MOA-ID Connector*: This connector forwards and transforms an authentication request to the Austrian national middleware MOA-ID (SPWare). Authentication responses from MOA-ID are also managed by this connector.
- *C-PEPS Connector*: The C-PEPS connector is the endpoint of the VIDP for outgoing and incoming messages to and from a C-PEPS. By the help of this connector, users originating from a PEPS country get the ability to authenticate at service providers supporting the MW model (MW-PEPS scenario –cf. Figure 3).

VI. Implementation and Deployment

Authentication may be seen as the most critical step in online processes. Secure implementation of authentication components are obviously essential, in particular in a project like STORK that aims at a European scale. This chapter describes the implementation, release and deployment of the common middleware architecture (VIDP) and discusses security and privacy aspects. The software implementation of the MW architecture based on J2EE [11] reference components developed by Austria and Germany is presented. Additionally, different deployment strategies are introduced. In the remainder of this chapter the security architecture is elaborated in more detail.

The main requirements for the chosen implementation strategy have been:

- Design of a dynamic and configurable model architecture
- Possibility of decoupling single modules for dynamic deployment
- Guaranteeing security on message and communication level
- Support of common database and application servers

A dynamic and configurable model architecture [21] allows for module configuration during runtime and flexible reactions in case of e.g. scalability bottlenecks. Configurations can easily be changed during runtime without requiring a server re-start [22].

The possibility of decoupling single modules allows flexible deployments. The VIDP implementation should allow easily coupling and decoupling of single modules during runtime. For example, it should be possible to add new SPWare Connectors and thereby supporting additional countries' eID tokens without re-starting the VIDP component. Additionally, components or modules can be updated during runtime [23]. Updates at runtime are important to avoid service interruptions. Note, that a VIDP hosts different country software (SPWare, PEPS Connectors). Release cycles do not allow for coordinated change management.

Security plays an important role in this architecture as personal data according to the EU Data Protection Directive [24] are transferred. We will discuss security and related privacy aspects in more detail in the separate sections VI.D and VI.E.

A more business oriented requirement is the support of different application and database servers. For MW countries a massive roll-out of the VIDP can be expected since this MW module will be installed in every service provider's domain if the support of foreign eIDs is desired. Due to the heterogeneity of the underlying infrastructures of service providers the VIDP should be deployable on various servers.

A. Development and Release Process

The development and release process for the implementation of the MW architecture is described in this section. The implementation of such a flexible MW architecture requires a thoroughly planned and structured development and release process. Therefore, the following objectives and processes have been set for the development and release process of the VIDP:

- Agile development and release process for easy extensibility and unexpected occurrences
- Secure development and release process
- Automated monitoring of the processes
- Secure and consistent configuration process
- A common development and release process for various infrastructures (e.g. application servers)

The development and release process is based on the concept of continuous integration [25]. By applying this concept, quality control is already guaranteed during the development process. We distinguish between three different phases or release levels of the VIDP components:

1. CI (Continuous Integration) for developers
2. QA (Quality Assurance) for testing
3. LIVE for the final release

In the CI phase, the developers of the individual VIDP components are responsible for testing the correct functionality. In the QA phase, the developers release a stable version to be tested by a quality assurance team. If all tests in the QA phase were successful a LIVE release can be build for distribution.

Besides the basic VIDP components, also the configuration of the individual modules requires an automated and traceable configuration process. In a first

[11] http://www.oracle.com/technetwork/java/javaee/tech/index.html

step, all configurations are taken into a staging system. After successful tests, the configuration settings are committed to the master system, which contains all main configurations and is back upped appropriately. The replication of the configurations for the individual release levels (CI, QA, LIVE) is handled by a so-called snapshoter. Global configuration settings are deployed on all systems whereas system-specific or contextual configurations are deployed on designated systems only.

B. *Implementation of the MW Architecture*

This section describes the actual implementation of the STORK middleware. To guarantee high flexibility and dynamics for the implementation, EJBs12 (Enterprise Java Beans) web services technologies had been chosen. Additionally, smooth interfaces were defined to allow flexibility for decoupling individual modules and dynamic deployment. Hence, adding or removing of modules during runtime does not negatively impact the system.

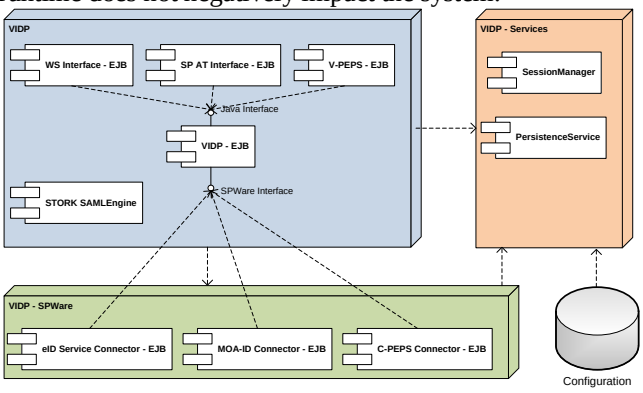

Figure 6. Component Diagram of the STORK Middleware

Figure 6 illustrates the component diagram of the implemented middleware architecture. To achieve great dynamism and flexibility the implementation has been split into three separate deployable modules:

- VIDP-Services
- VIDP-SPWare
- VIDP

The VIDP-Services module is responsible for general or support tasks, e.g. managing authentication sessions or handling the communication with the external database. The database holds all required configuration information for the individual modules and components.

The VIDP-SPWare module contains the country-specific SPWare Connector components. These connectors handle the communication with the national MW module (SPWare). The C-PEPS Connector component constitutes a special component as it manages the communication with foreign C-PEPSs if a particular country relies on the PEPS and not the MW model. All connectors are modeled as EJBs. For configurations, the VIDP-SPWare module accesses the VIDP-Services module.

The VIDP module constitutes the main module of the MW implementation. The routing functionality is implemented in the VIDP-EJB component. The service provider specific authentication interfaces are also modeled as EJB components. The national MW connector modules (SPWare Connectors) are included in the VIDP-SPWare module and thus the VIDP only connects to them. The separate STORK SAMLEngine component handles all tasks relating to the common STORK interface protocol which is based on SAML. Again, for configurations also the VIDP module relies on the VIDP-Services module.

Because of the separation of the VIDP functionality into different modules, also different deployment options exist.

C. *Deployment Options*

The middleware implementation shown in Figure 6 allows a flexible arrangement of the modules for deployment. Depending on availability of resources or other desired properties such as flexibility or maintenance efforts, different deployment strategies can be chosen. Moreover, static or dynamic extensibility of the VIDP is supported. In this context, the term dynamic means that modules (e.g. C-PEPS Connector, VIDP-SPWare) can easily be added or removed during runtime without negatively influencing the complete VIDP operation.

The following deployment opportunities are supported:

- Coupled Deployment
- Loose Deployment

When choosing a coupled deployment, all VIDP modules (VIDP-Services, VIDP-SPWare, VIDP) are deployed on a common server instance. The advantage of this approach is that all modules reside on the same machine which gives less maintenance effort but less flexibility and performance.

Within a loose deployment model, the VIDP modules such as VIDP-Services or VIDP-SPWare can be deployed individually as single and distributed instances. This increases flexibility in case of performance and scalability bottlenecks. Nevertheless, the distribution of components raises the risk that components may be inaccessible because of network errors or shutdowns [26].

To support this diversity of flexible and scalable deployment approaches, APIs based on the J2EE-Interfaces Local, Remote and Web Services (SOAP) for the individual modules had been defined. The transition from one interface to another one (e.g. from Remote to Web Services) can easily and dynamically be carried out during runtime without interfering the operation of the respective module.

D. *Security*

Security plays a major role in the STORK context as well as in its framework implementations. Personal data of

[12] http://www.oracle.com/technetwork/java/index-jsp-140203.html

EU citizens are transmitted across borders, are processed, and are temporarily stored. These personal data define valuable assets, which must be protected. STORK had a dedicated security team that defined security requirements and principles [27]. These have as well been implemented by the VIDP. The security principles follow a threat – objective – security function approach: A threat analysis has been carried out. Threats include impersonation or a possible loss of confidentiality, integrity, or availability of personal data identified. These threats lead to security objectives that need to be met by security functions. These security functions must be implemented by the individual STORK components or modules. A selection of these security functions and their implementation in the VIDP are described in the next sub-sections.

The interfaces between entities or components define the critical parts where impersonation or a loss of security can occur. Figure 7 illustrates the critical interfaces of the VIDP which must be especially protected.

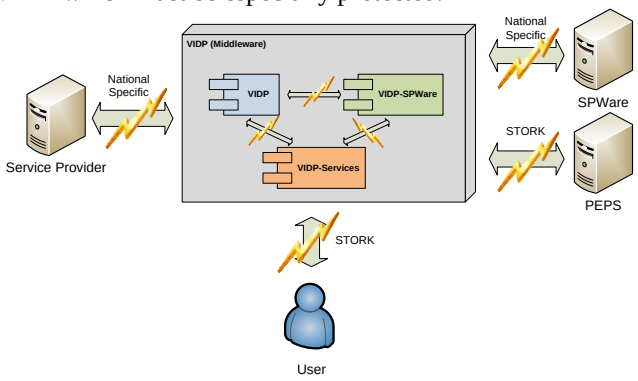

Figure 7. VIDP critical interfaces

Critical interfaces can be identified internally and externally to the VIDP. The protection of internal interfaces is especially important if a loose deployment option is preferred, where the VIDP implementation components (VIDP-Services, VIDP-SPWare, VIDP) are distributed. The external interfaces must be protected in every situation where an external entity of the VIDP (e.g. Service Provider or PEPS) is involved. In other words, whenever personal data leave the VIDP and are transferred to another entity the data must appropriately be protected. In the following we describe how these external and internal interfaces were secured.

1) VIDP External Interfaces

This sub-section identifies the critical external interfaces of the VIDP and shows how the predefined security requirements of STORK were met.

SP ⇔ VIDP Interface:

Via this interface data are transferred between a national service provider and the VIDP. In the MW model, the general idea is that the VIDP is directly installed in the SP domain to enable end-to-end security between the user and the service provider. Thus, there are no further security requirements that must be fulfilled for the VIDP except for the SP itself. In fact, the VIDP can be seen as being a part of the SP. However, the SP has to ensure that the internal SP-VIDP connection is secured properly.

In case this SP-VIDP interface is externalized, the VIDP needs to support the security functions of the national specific service provider interface and its protocol. The current VIDP implementation supports national SP interfaces of Austria and Germany. The connection between an Austrian SP interface and the VIDP is secured by the use of TLS/SSL certificates. The German SP interface is Web service-based and requires a mutually secured and authenticated TLS communication channel.

VIDP ⇔ SPWare Interface:

Identification and authentication data are exchanged between the VIDP and the national MW module (SPWare) through this interface. According to the main idea of the MW model, all supported national MW modules are installed close to the VIDP within the SP domain. Hence, this interface can be assumed as SP internal interface which does not require higher protection than the SP domain itself. However, in case of externalization of this interface (as illustrated in Figure 7) the data passing through must be appropriately protected. Similar to the SP-VIDP interface, the current VIDP implementation supports connections to the Austrian and German national MW module. Both countries rely on a mutually authenticated TLS communication channel for data transfer between the VIDP and the SPWare.

VIDP ⇔ PEPS Interface:

This interface implemented by the VIDP relies on the common STORK interface specification [18] and its protocol. The common STORK protocol is used for the secure data transfer between a VIDP and a PEPS. Since this protocol bases on SAML 2.0 also all security related functionality is aligned to this well-established standard. In particular, for data transfer between STORK entities the SAML Web SSO Profile [28] with the HTTP Post Binding [29] is used. Thereby, all in- and outgoing messages must be properly digitally signed using the XML-DSig syntax [30]. Digital signatures ensure message integrity, non-repudiation and authenticity. Authenticity can be guaranteed because only digital certificates issued for STORK entities are trusted.

To further improve security, the STORK specification allows to encrypt parts (especially user data) of the transmitted messages. For encrypting such parts, the XML Encryption syntax [31] can be used. In addition, instead of the SAML Web SSO Profile the SAML Holder-of-Key (HoK) Profile [32] may be used. This profile ensures a stronger authentication and security context between the identifying and authenticating provider, the service provider, and the user's client. This higher strength is based on client's presentation of the same X.509 certificate, which results from the TLS handshake, to both providers. However, the HoK Profile is currently not widely adopted in standard components, e.g. Web browsers.

VIDP ⇔ User Interface:

Through this interface, required interactions between the user and the VIDP are handled. The user accesses this interface by a standard Web browser. To guarantee a high level of security, all connections to the VIDP are secured by the use of TLS/SSL. Users are able to verify the authenticity of the VIDP by checking the corresponding X.509 certificate.

In general, users are not required to enter any data into a Web page or form presented by the VIDP. However, all input messages or input data are validated by the VIDP against syntax, range, length, etc. to prevent e.g. cross-site scripting attacks. Additionally, during the implementation and testing phase the developers considered several Web application security issues, especially the ones presented by the OWASP [33].

2) VIDP Internal Interfaces

The VIDP internal interfaces constitute those interfaces between the three VIDP implementation components (VIDP-Services, VIDP-SPWare, VIDP). For the VIDP internal interfaces security issues only come into play if a loose deployment option for the VIDP is chosen. In this deployment option, the VIDP implementation components can be deployed remotely and distributed for achieving higher flexibility and scalability.

For implementing the VIDP the EJB technology has been chosen. This shifts application security aspects to the server implementation hosting the VIDP [34]. This simplification holds especially for a coupled deployment of the VIDP individual components, but it cannot be relied on when applying a distributed (loose) deployment model. To achieve the same level of security independent of the deployment option, so-called security gateways were implemented protecting the remote communication between the three VIDP implementation components.

Those security gateways are modular available and are responsible and were especially designed for supporting individual security functions such as authentication and authorization, signature or encryption services, or preventing denial-of-service (DOS) attacks. Authentication between components is based on mutual SSL/TLS authentication. For authorization between the individual components the well-known Role Based Access Control (RBAC) models [35] and Attribute Based Access Control (ABAC) models [36] are supported. Again, for signature and encryption functionality the XML-DSig and the XML-Enc standard had been chosen. The DOS protection security gateway only allows a maximum number of requests to a VIDP implementation component during a certain time frame. In addition to these security service gateways, gateways supporting supplementary functionality such as schema validation or message logging for auditing purposes had been implemented.

E. Privacy

Most of the data processed within the STORK environment are personal data according to the EU Data Protection Directive [24]. This section discusses some fundamental privacy principles and furthermore how these were tackled by the VIDP implementation. The following privacy-preserving principles were considered:

- Exchange of national identifiers
- Minimum disclosure principle for personal attributes
- User centricity
- Data unlinkability

Article 8 (7) of the data protection directive states that *"Member States shall determine the conditions under which a national identification number […] may be processed"* [24]. This article has been implemented individually by each EU Member State into national law. What several implementations of the directive have in common is that the use of unique identifiers is restricted. A consequence is that cross-border use is not possible in many cases. To overcome that situation, the STORK framework and its implementation supports the calculation of transient identifiers. Such transient identifiers can be generated using one-way hash algorithms (e.g. the SHA family [37]) by deriving the unique identifier for a specific country, specific sector, or specific application. Such calculations are also supported by the VIDP implementation. In fact, context-specific identifiers are a core privacy function of both the Austrian and the German eID system – the supporters of the MW model.

The minimum disclosure principle specifies that only a relevant amount of personal data must be processed. Article 6 (1) (c) of the EU Data Protection Directive states that personal data must be *"equate, relevant and not excessive in relation to the purposes for which they are collected and/or further processed"* [24]. STORK follows this principle and only allows the transfer of data which are really required. Service providers can request mandatory or optional attributes and users can allow or deny the personal attribute transfer.

User centricity within the STORK context means that users can always control how their personal data are obtained and how the data are transferred. This requirement is fulfilled by asking the user's consent for data transfer and data processing. Consenting defines a fundamental requirement of the EU Data Protection Directive and is stipulated in Article 7 (a). For the VIDP, the process of consenting is actually individually implemented by each national MW module.

Unlinkability refers to a property that data shall not be shared unless the user consents or such sharing is legitimate. It defines one major privacy principle within STORK. Data linkage or even profiling of users mostly takes place if central services are involved. Since there does not exist a

central instance of the VIDP this privacy requirement can be easily fulfilled by the MW model. Context-specific identifiers that are specific for a service provider prevent from linking to a user's account at other service providers.

VII. Conclusions

We presented a secure identification and authentication architecture (Virtual Identity Provider – VIDP) that is based on the so-called middleware (MW) approach of STORK. This architecture supports cross-border identification and authentication of different eIDs of various EU Member States. In general, the STORK project defines two basic approaches for national eID infrastructure interoperability, the MW approach and the PEPS approach. In comparison to the PEPS model, main advantages of the MW approach are end-to-end security and liability as there is no intermediary between the user and the service provider.

The MW architecture has been developed by Austria and Germany and was implemented based on J2EE components. Thereby, emphasis lay on dynamic configurations, dynamic deployment, security, and the support of popular database and application servers. The development and release process was aligned according to these requirements and therefore common and popular software modules were used. The implementation has been tested in the six STORK pilot applications. These are productive environments, such as national eGovernment portals, the STORK e-Delivery pilot [38], the STORK Safer Chat pilot [39], the STORK Student Mobility pilot [40], or ECAS – the central authentication service of the European Commission.

STORK was a success, as cross-border acceptance of national eIDs could be successfully demonstrated in real environments. However, while STORK showed that cross-border eID interoperability is technically feasible, some hindering issues still remain open for future investigations. Issues on organizational level are for example the mapping of personal identifiers from national registers between countries. Another issue is harmonization of legislation as e.g. eID registration procedures vary between countries. Furthermore, in terms of acceptance of individual credentials for authentication still some work needs to be done. For instance, to qualitatively ensure the authentication levels proposed by STORK some independent auditing and validation procedures would be required. [41]

Nevertheless, the STORK results on cross-border eID for natural persons are taken up by a follow-up project STORK 2.0 [13] that further elaborates on mandates and representation, such as representing a legal person. The lessons learned, in particular that technology is not the hindering factor of cross-border eID, but the lacking trust framework such as lacking mutual recognition, influenced ongoing European policy measures. The main is a proposed European Community legal framework for eID [1] Furthermore, the gained experience of developing STORK could also have implications for other evolving efforts such as NSTIC.

[13] http://www.eid-stork2.eu

References

[1] European Commission: Proposal for a Regulation of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market, COM(2012) 238/2.

[2] European Union: Ministerial Declaration, Manchester, United Kingdom, on 24 November 2005

[3] European Commission: Ministerial Declaration on eGovernment approved unanimously, Malmö, 18 November 2009

[4] European Union: Directive 2006/123/EC of the European Parliament and of the Council of 12 December 2006 on services in the internal market

[5] European Commission - IDABC. 2009. *eID Interoperability for PEGS: Update of Country Profiles*.

[6] Palfrey, J., Gasser, U.: *Digital Identity Interoperability and eInnovation*, Case Study. Berkman Publication Series (2007)

[7] Bauer, M., Meints, M., and Hansen, M. 2005. *D3.1: Structured Overview on Prototypes and Concepts of Identity Management* System, FIDIS

[8] Jøsang, A., Al Zomai, M., & Suriadi, S. (2007). Usability and privacy in identity management architectures. In *ACSW '07 Proceedings of the fifth Australasian symposium on ACSW frontiers* (pp. 143-152). Darlinghurst, Australia.

[9] Jøsang, A., and Pope, S. 2005. User centric identity management. In *AusCERT Asia Pacific Information Technology* (pp. 1-13).

[10] Alpár, G., Hoepman, J.-H., and Siljee, J. 2011. *The Identity Crisis - Security, Privacy and Usability Issues in Identity Management*.eprint

[11] Neuman, C., Yu, T., Hartman, S. and Raeburn, K. 2005. The Kerberos Network Authentication Service (V5). RFC 4120. Internet Engineering Task Force (IETF)

[12] Lockhart, H. and Campbell, B. 2008. *Security Assertion Markup Language (SAML) V2.0 Technical Overview*. OASIS Committee Draft 02.

[13] Kaler, C. and McIntosh, M. 2009. *Web Services Federation Language (WS-Federation) Version 1.2*. OASIS Standard.

[14] Nadalin, A., Kaler, C., Monzillo, R., and Hallam-Baker, P., 2006. *Web Services Security: SOAP Message Security 1.1 (WS-Security 2004)*. OASIS Standard Specification

[15] The White House. 2011. National Strategy for Trusted Identities in Cyberspace (NSTIC)

[16] European Commission - MODINIS. 2006. *The Status of Identity Management in European eGovernment initiatives*.

[17] Siddhartha, A. 2008. *National e-ID card schemes: A European overview*. Information Security Tech. Report. 13, 2 (May 2008), 46-53

[18] Alcalde-Morano, J., Hernández-Ardieta, J.L., Johnston, A., Martinez, D., Zwattendorfer, B., and Stern, M. 2011. *D5.8.3b Interface Specification*. STORK Deliverable

[19] Leitold, H., Hollosi, A., and Posch, R. 2002. Security Architecture of the Austrian Citizen Card Concept. In *Proceedings of the 18th Annual Computer Security Applications Conference* (2002)

[20] Ferderal Office for Information Security (BSI). 2009. *eCard-API-Framework* (BSI TR-03112)

[21] Allen, R., Douence, R., & Garlan, D. 1997. Specifying dynamism in software architectures. *Proceedings of Foundations of Component-Based Systems Workshop* (pp. 1-12).

[22] Kramer, J., and Magee, J. 1985. Dynamic Configuration for Distributed Systems. *IEEE Transactions on Software Engineering*, SE-11(4), 424-436.

[23] Ritzau, T., and Andersson, J. 2000. Dynamic deployment of Java applications. In *Java for Embedded Systems Workshop*.

[24] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data

[25] Humble, J., and Farley, D. 2011. Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Addison-Wesley

[26] Paal, S., Kammüller, R., and Freisleben, B. 2003. Separating the concerns of distributed deployment and dynamic composition. In *On the move to meaningful internet systems* (pp. 1292-1311).

[27] Stern, M. 2011. *D5.8.3d Security Principles and Best Practices*. STORK Deliverable

[28] Hughes, J., Cantor, S., Hodges, J., Hirsch, J., Mishra, P., Philpott, R., and Maler, E. 2009. *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS.

[29] Cantor, S., Hirsch, F., Kemp, J., Philpott, R., and Maler, E. 2009. *Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS

[30] Bartel, M., Boyer, J., Fox, B., LaMacchia, B., and Simon, E. 2008. *XML Signature Syntax and Processing (Second Edition)*. W3C Recommendation.

[31] Imamura, T., Dillaway, B., and Simon, E.2002. *XML Encryption Syntax and Processing*. W3C Recommendation.

[32] Lockhart, H. and Hardjono, D. 2010. *SAML V2.0 Holder-of-Key Web Browser SSO Profile Version 1.0*. OASIS Committee Specification 02.

[33] The Open Web Application Security Project (OWASP). 2010. *OWASP Top 10 - 2010 – The Ten Most Critical Web Application Security Risks*.

[34] Roman, E., Patel Sriganesh, R., and Brose, G. 2005. Mastering Enterprise JavaBeans. Third Edition. Wiley Publishing

[35] Sandhu, R., Coyne, E., & Feinstein, H. 1996. *Role-based access control models*. Computer, 29(2), 38- 47.

[36] Yuan, E., & Tong, J. 2005. Attributed based access control (ABAC) for Web services. In *International Conference on Web Services, 2005*. ICWS 2005.

[37] National Institute of Standards and Technology (NIST). 2002. *Secure Hash Standard*. Federal Information Processing Standards Publication 180-2

[38] Tauber, A., Zwattendorfer, B., Zefferer, T.: STORK: Pilot 4 Towards Cross-border Electronic Delivery. In: Electronic Government and Electronic Participation - Joint Proceedings of IFIP EGOV and ePart 2011

[39] Knall, T., Tauber, A., Zefferer, T., Zwattendorfer, B., Axfjord, A., Bjarnason, H.: Secure and Privacy-preserving Cross-border Authentication: the STORK Pilot "SaferChat". Proceedings of the Conference on Electronic Government and the Information Systems Perspective (EGOVIS 2011)

[40] Berbecaru, D., Lioy, A., Mezzalama, M., Santiano, G., Venuto, E., and Oreglia, M. 2011. Federating e-identities across Europe, or how to build cross-border e-services. In *AICA 2011: Smart Tech and Smart Innovation conference*. (pp. 1-10)

[41] Koulolias, V., Kountzeris, A., Leitold, H., Zwattendorfer, B., Crespo, A., Stern, M. , "STORK e-privacy and security," In 5th Internationl Conference on Network and System Security (NSS) 2011, pp.234-238

BIOGRAPHIES

Bernd Zwattendorfer has studied Telematics and received his master's degree from Graz University of Technology. Additionally, he holds a master's degree in International Business received from University of Graz. In 2007 he joined the eGovernment Innovation Center (EGIZ) in Graz, which supports the Austrian Federal Chancellery in further developing the Austrian ICT-Strategy by research and innovation. He is currently working on several topics related to IT security and eGovernment, focusing on electronic identity and cloud computing. During his work he participated in the following EU projects: FP6 project eGovBus, LSP STORK, GINI-SA.

Ivo Sumelong holds a master's degree from University of Duisburg-Essen in Information and Communication Engineering. He gained experience as software engineer at various companies since many years. Currently, he is senior software architect at OpenLimit SignCubes GmbH working on several eID related projects.

Herbert Leitold received his masters in telecommunication and informatics in 1996. He has been research assistant at Graz University of Technology from 1996 to 2001. Since 2001 he is site manager of the Secure Information Technology Center – Austria (A-SIT) technology assessment group. From 2005 to 2012 he has been head of the E-Government Innovation Center (EGIZ). Herbert Leitold participated in several European research projects such as the LSPs STORK, STORK 2.0, and e-SENS.