SECURE CLOUD AUTHENTICATION USING EIDS

Bernd Zwattendorfer, Arne Tauber

E-Government Innovation Center (EGIZ),
Graz University of Technology, Graz 8010, Austria
bernd.zwattendorfer@egiz.gv.at, arne.tauber@egiz.gv.at

Abstract: Identification and authentication are essential security functions for regulating access to protected data. Considering that, most cloud service providers rely on weak authentication mechanisms such as username/password schemes. While username/password authentication may be sufficient for simple customized applications, cloud applications in more sensitive areas such as in e-Government require more reliable and secure mechanisms. We close this gap for such cloud applications by applying the STORK framework for secure cloud authentication using eIDs. The STORK framework supports various national eID solutions and will be the relevant eID framework across Europe in future. We demonstrated our approach by enabling eID authentication at two selected public cloud service providers. Finally, we also moved the STORK framework to the cloud to apply the full cloud computing paradigm.

Keywords: cloud computing; authentication; identification; electronic identity; eID

1 Introduction

Cloud service providers offer their cloud services based on different models, e.g. the *Infrastructure, Platform, or Software as a Service* model (IaaS, PaaS, SaaS). When applying these models, several requirements must be considered and met, especially in the field of security. Two important security requirements are user identification and authentication. Many cloud service providers still rely on simple username/password schemes although they early turned out to be weak [1]. While username/password schemes may be sufficient for protecting simple customized services, cloud applications e.g. providing e-Government services must rely on stronger and qualified authentication mechanisms for being compliant with data protection regulations or national law. Such stronger authentication mechanisms are e.g. smart cards or mobile phones based on two factor authentication. A lot of countries have already rolled-out national eID solutions supporting strong authentication and unique qualified identification. Unique qualified identification is particularly essential within sensitive areas such as e-Government, especially if there are a high numbers of users such as the population of a whole state.

In this paper we combine the topics of cloud computing and qualified identification and authentication. We discuss how highly secure national eID solutions can be used for unique qualified identification and authentication at different cloud service providers. This offers cloud service providers the possibility to penetrate market areas where higher security requirements for identification and authentication must be met (e.g. the e-Government or e-Health sector). For achieving this, we extended the modular and scalable eID interoperability architecture proposed by the STORK[1] (Secure identity acRoss boRders linKed). This architecture has been developed together by 18 EU Member States within the STORK project and supports cross-border eID interoperability of various national identity systems. This framework constitutes the dominant eID system to be adopted within Europe in future. Sustainability of this framework is further given by the ISA[2] programme, which takes over maintenance tasks of the common specifications and software modules, and the successor project STORK 2.0[3], which aims on more widespread piloting of the framework and cross-border interoperability of legal persons.

The paper is structured as follows. First, we briefly introduce different types of cloud identity models and overview related work in the field of identity management (section 2 and 3). Section 4 is dedicated to the STORK framework and its architecture. Section 5 finally explains how eID authentication was successfully achieved at two selected public cloud providers applying the STORK architecture. At the end, conclusions are drawn.

2 Cloud Identity Models

Identification and authentication are no new issues, hence several identity models have already emerged [2]. Most models in traditional Web-based systems follow the identity triangle architecture, involving a user, an identity provider (IdP), and one or more service providers (SP) in the communication process. For instance, Palfrey and Gasser [3] distinguish three different types of identity models (user-centric model, central model, and federated model). Distinction criterion however is the location where identity data are stored.

Since identification and authentication are not less important in the area of cloud computing, new cloud identity models tailored to the needs of cloud computing can be defined. For example, Goulding classifies such

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

[2] http://ec.europa.eu/isa

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

models in his whitepaper [4]. In the following, we describe three different cloud identity models based on the work of Goulding. In the remainder of our paper we use this classification to refer to and to see where our approach can be classified in.

2.1 Identity in the Cloud

The *Identity in the Cloud* model refers to the architecture, where the service provider itself also acts as identity provider. The cloud service provider hosting the cloud application carries out the authentication process and additionally stores all required identity data. Hence, identity data are *in the cloud*. This model matches the central identity model of [3], defining the special case where service and identity provider merge. Typical examples for such providers are Google or Salesforce.com who are offering SaaS including their identity management system. One main advantage of this model is that organizations do not need to maintain a separate identity management system since all user profiles and accounts are directly offered and managed by the cloud service provider. Nevertheless, existing user accounts cannot be easily migrated.

2.2 Identity to the Cloud

This cloud identity model is similar to the central identity model defined by [3], where the identification and authentication processes are delegated from the cloud service provider, which is hosting the application, to an external identity provider. The identity provider stays in the domain and under control of the enterprise or organization[4] and thus identities are provided *to the cloud*. In this model, the identity provider is fully responsible for the user management. The cloud service provider just receives authenticated user and identity data from the identity provider. The advantage of this model is that existing identity management systems can be re-used and no migration to the cloud is required. Several cloud service providers offer specific interfaces for the usage of external identity providers. By the help of this interface identities can also be federated, matching the federated approach described by [3].

2.3 Identity from the Cloud

In this service model identities are provided *from* an identity provider in *the cloud*. This means that both the service and identity provider are operated in the cloud. In contrast to the *Identity in the Cloud* model (see section 2.1), the identity and service provider can but need not be operated by the same cloud service provider. This model can be also seen as an *Identity as a Service* cloud model [5]. In this model organizations can choose their desired operator for their identity management based on specific requirements (e.g. data protection regulations that force the storage of identity data in a specific region). Outsourcing of the identity management system to the cloud saves costs and maintenance efforts for the organization. For instance, account provisioning or auditing can be fully automated with minimal changes for the organization [6].

3 Related Work

Various identity management systems already exist and have evolved. In this section we briefly introduce a couple of identity management systems and standards which gained importance over the past years.

One of the first protocols enabling identification and authentication across domains was Kerberos. Kerberos allows for secure authentication in unsecure TCP/IP networks. On application level, one of the first adopters introducing a central authentication system in the WWW was Microsoft with Microsoft Passport (latterly called Windows LiveID). Relying on a decentralized architecture, the Liberty Alliance Project (latterly integrated into the Kantara initiative[5]) and Shibboleth[6] entered the market supporting identity federation and single sign-on. Both projects heavily influenced the development of version 2.0 of SAML[7]. SAML currently defines one of the most important and adopted standards in the field of single sign-on and identity federation. A similar framework enabling federation constitutes WS-Federation, which is part of the WS-* framework. Another lightweight alternative offering decentralized authentication is OpenID[8], which has already penetrated the identity market in the last couple of years. In addition, OAuth[9] as a REST-based standard is widely used (e.g. in social networks such as Facebook) enabling besides authentication authorization as well. SAML or OAuth are already employed by several cloud service providers [6], but also OpenID or WS-Federation are promising identity and authentication frameworks for cloud computing adoption.

Secure authentication and identification are also inevitable processes for governments. The USA currently has ambitions on creating a secure identity system. This aim has been published in the National Strategy for Trusted Identities in Cyberspace (NSTIC) in 2011 and should facilitate access to public and private sector services. Moreover, especially in Europe several countries have already rolled-out national eID solutions supporting e-Government and e-Business applications. Some countries rely on user-centric approaches based on smart-cards or mobile phones, others federate identities between authentication gateways. However, most countries rely on a Public Key Infrastructure (PKI) and the X.509 standard for their national eID solutions. A thorough overview of various national eID solutions can be found in the IDABC eID country reports [7].

[4] The identity provider could also be deployed in a private cloud, but we consider this constellation in the *Identity from the Cloud* model.

[5] http://kantarainitiative.org

[6] http://shibboleth.net

[7] http://saml.xml.org/

[8] http://openid.net

[9] http://oauth.net

The various national eID solutions are very heterogeneous across Europe. They differ on technological, operational and legal level. Due to that, in 2008 the European Commission has launched the European large scale pilot project STORK, which involved 18 EU Member States. STORK aimed on achieving cross-border eID interoperability between various national eID systems. The project has finished at the end of 2011 and successfully demonstrated and piloted cross-border eID federation between the participating countries. The results of this project will be amended and extended by its successor project STORK 2.0, which additionally focuses on sustainability and furthermore on interoperability of legal persons across Europe. STORK is currently heavily pushed by the European Commission and will be the relevant eID framework across Europe in future. This foreseeable future relevance and the great support of a variety of national eID solutions were the main arguments for taking and adapting the STORK framework also for secure cloud authentication.

4 STORK Architecture

Within the STORK project, a secure and reliable eID interoperability framework and architecture has been specified and developed. This framework allows for secure cross-border identification and authentication by using the own national eID. In other words, this means that EU citizens are capable of authenticating at foreign online applications using the eID issued by their home country. This objective was a fundamental principle of the STORK project. No new EU-wide solution should be introduced but furthermore existing national eID concepts should be made interoperable. This aim was achieved by building an appropriate interoperability layer on top of those individual national solutions.

Basically, the framework consists of a combination of two distinct identity models varying across the Member States. The first model is a proxy-based approach based on intermediaries (PEPS Model). This model can be seen as a representative of the federated model defined by [3]. The second identity model used in STORK is the so-called Middleware (MW) model. In this model users are directly authenticated at the service provider without any intermediary. This model can be seen as user-centric identity model according to [3]. For applying the MW model, the service provider installs a so-called VIDP (Virtual Identity Provider) in its domain, which supports all desired national authentication methods, irrespective of the underlying identity model (PEPS or MW model). A more detailed description on the STORK architecture and the individual interoperability models (PEPS and MW model) can be found in [8]. The communication between STORK entities is carried out via a well-specified protocol [9], which is SAML-based.

For our approach, using the STORK framework for secure cloud authentication, we rely on the VIDP architecture because of its modular design. The VIDP architecture developed in STORK supports a variety of national eID systems, irrespective of the underlying identity model (PEPS or MW model). Thereby, a common component can be extended by several modules. These modules either support national authentication protocols for service providers or manage the communication with national eID systems. Figure 1 illustrates this modular VIDP architecture.

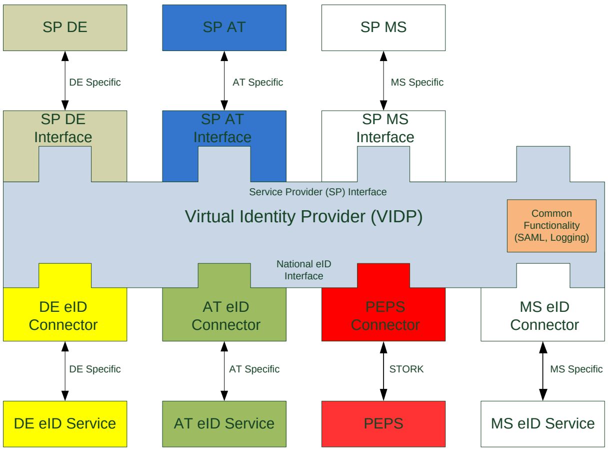

Figure 1 VIDP Architecture

The so-called plug-ons implement the *Service Provider Interface* of the VIDP and constitute Member State specific endpoints for receiving authentication requests. These authentication requests are further processed by the VIDP and delegated to the respective national eID module. The national eID modules are responsible for carrying out the actual identification and authentication process of the user. Connectors to these national eID modules implement the *National eID Interface* and are named plug-ins. A special plug-in is the PEPS Connector module. The PEPS Connector module allows for interconnecting with other countries which follow the PEPS model in their domestic domain. In Figure 1, the specific interface implementations (both plug-ins and plug-ons) for Austria (AT) and Germany (DE) are illustrated. The PEPS Connector covers the other STORK countries enabling cross-border eID federation. The modular architecture of the VIDP allows for adding arbitrary MS specific eID systems by simply adding appropriate connector modules. Hereby both identity model types defined in STORK (PEPS or MW model) are supported. In addition, arbitrary SP interface modules can be added, enabling different authentication protocol support.

5 EID-based Cloud Authentication

Unique identification and secure authentication are essential processes when securing cloud applications. Currently, most cloud computing service providers rely on simple username/password schemes that early turned out to be weak [1]. This section explains how - and by relying on the STORK framework - more sophisticated authentication mechanism based on eIDs can be used for more secure and reliable identification and authentication at applications offered by cloud service providers. For achieving that, we considered the *Identity* *to the Cloud* model as described in section 2.2 in a first phase. This model allows for more control and privacy protection than the other models and hence for being compliant with most national regulations and laws. In a second phase, we decided to apply the *Identity from the Cloud* model (cf. section 2.3) to fully support the cloud paradigm of an Identity as a Service model using eIDs. In the following sub-sections, we describe the first approach and latterly the second.

5.1 Identity to the Cloud

For our proof of concept applying this identity model, we selected two public cloud service providers (Google and Salesforce.com) for demonstrating secure eID identification and authentication at their SaaS applications. Both providers provide single sign-on interfaces for external eID federation. One protocol they both rely on is SAML. Although both rely on SAML, the individual specifications and implementations are different. Google offers the SSO interface based on SAML 2.0, Salesforce.com provides external authentication capabilities for the SAML versions 2.0 and 1.1. Google supports the SAML AuthnRequest/Response protocol whereas Saleforce.com does not require the use of a specified SAML protocol, which allows the reception of unsolicited response messages. However, both providers rely on the SAML HTTP Post Binding for SAML message transfer. Besides SAML, both providers also offer other external authentication possibilities such as OAuth or OpenID. The latter is only supported by Google.

As identity provider for authentication at these public cloud service providers we took the VIDP from STORK because of its flexible and modular architecture. On the one hand, various national eID systems are already supported and covered by this architecture. On the other hand, the pluggable design allows for easy integration of new service provider authentication interfaces as required for the communication with Google and Salesforce.com. The modular VIDP architecture allows easy extensibility, hence also other cloud service providers offering different interfaces could be supported.

To allow for secure eID authentication at Google Apps and Salesforce.com, we enhanced the VIDP architecture by adding two new service provider modules. We implemented two additional modules supporting the SSO interface of the individual cloud service provider on the one side. On the other side, those modules furthermore implement the *Service Provider Interface* and are plugged on the VIDP. Both modules are capable of receiving SAML-based authentication request and response messages carrying identity and authentication information. We decided using SAML 2.0 as cloud authentication protocol for both providers because basic SAML functionality was already available by the VIDP implementation. Figure 2 illustrates this extended VIDP architecture.

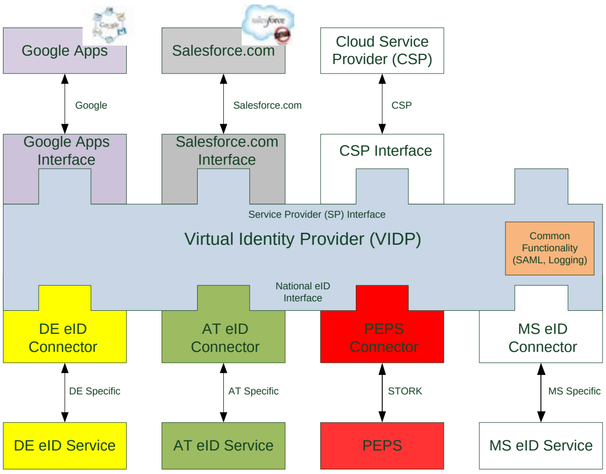

Figure 2 Extended VIDP architecture supporting eID-based cloud authentication

Although both Google and Salesforce.com rely on SAML 2.0, during the implementation a different behavior of the SAML interfaces could be determined. Hence, two different SAML-based VIDP extension modules were required. Although Google provides some documentation of the SSO interface on their web page, this information is rather out of date. For instance, they announce that the downloadable reference code for external authentication at Google Apps is not compatible with the current implementation of the SSO interface. Furthermore, the SAML message sent to the SSO interface must perfectly match the requirements of Google, which are weakly documented. This means that additional SAML elements or attributes included in the SAML message resulted in unsuccessful authentication attempts, even if those additional elements were conformant with the SAML 2.0 specifications. For message authentication and data integrity the SAML messages need to be signed and for verification the digital certificate needs to be uploaded to the Google services. However, Google seems to simply compare the received and stored certificate instead of doing real certificate revocation checking.

In comparison to Google, Salesforce.com provides a more detailed documentation on the SAML-based external SSO interface. Moreover, Salesforce.com offers an online validation service where SAML messages can be approved for Salesforce.com applicability. This enables a faster and more reliable error analysis. Salesfore.com is also not that strict if unknown SAML elements or attributes are included in the SAML message, as long as they are conformant with the SAML specification. In addition, due to the lack of a SAML protocol support Salesforce.com allows IdP-initiated authentications only.

Besides secure authentication, unique identification plays a major role in governmental processes; hence many countries also uniquely identify their citizens based on special identifiers. How citizens are really electronically identified depends on several aspects (e.g. organizational or legal requirements), thus the identification process varies across countries. For instance, some countries rely on persistence identifiers whereas others prefer a sectoral model. Due to this heterogeneity of identifiers and for being compliant with most privacy regulations, we decided to apply the sectoral model also for our implementation. Therefore, we securely derive the identifier received from the national eID system before transmitting it to the cloud service provider. The identifier is derived using cryptographic one-way hash functions. This means that a separate unique identifier is generated for both providers enabling privacy protection and making user tracking or profiling impossible[10]. This identifier is further used for identity federation at the respective cloud service provider.

In more detail, the identifier to be used for authentication at Google must follow the format of an e-mail address. The derived identifier was transformed in the format *…@xyz.com*, where *xyz.com* denotes the custom domain used for Google Apps. For successful authentication at Google Apps, this identifier must be registered before. No authentication without prior registration is possible. In contrast to that, Salesforce.com allows seamless and on-the-fly user registration. For identification either the username internally used by Salesforce.com or an identifier to be federated with the Salesforce.com account can be used. In our implementation we favored the second approach as no format changes of the generated derived identifier are required.

5.2 Identity from the Cloud

After successful implementation of the *Identity to the Cloud* model, we decided to further try moving the extended VIDP architecture also into the cloud for applying the *Identity from the Cloud* model. Thereby, we could fully support the cloud paradigm of an *Identity as a Service* model using eIDs. For deploying the VIDP in the cloud we selected Jelastic[11] and the Google App Engine[12] as underlying Java-based cloud platforms. We early succeeded on the deployment on Jelastic, as no code modifications were required. Jelastic supports a complete Java virtual machine (JVM) and various application servers in its provisioned cloud platforms, where the VIDP could be deployed to. The deployment of the VIDP on the Google App Engine was even more difficult, as the provided JVM of the Google cloud platform provides a limited subset of functionality only. Hence, several changes on the code were necessary. For instance, code snippets writing to the file system, running threads, or raw network sockets, which are all not supported by the Google App Engine, had to be exchanged and re-written to use only functionality supported by this platform. Nevertheless, we also succeeded in deploying the extended VIDP on this cloud platform to apply an Identity as a Service model.

6 Conclusions

Many cloud providers still rely on unsecure mechanisms (e.g. username/password schemes) for user authentication at cloud applications. Especially in complex and sensitive areas such as e-Government these mechanisms reach their limits. To bypass this gap, we showed how various national eID systems can be used for secure cloud authentication and unique user identification. We therefore extended the STORK VIDP architecture to support external eID authentication at different cloud service providers. We demonstrated its applicability by securely authenticating at two different public cloud service providers (Google and Salesforce.com) using various national eIDs. The proposed approach follows the *Identity to the Cloud* model as explained in section 2.2. Furthermore, we additionally tried to move the enhanced VIDP into the cloud for applying the *Identity from the Cloud* model (cf. section 2.3). While this was technically feasible - we succeed in deployment on the two public cloud platforms Jelastic and Google App Engine – this might contradict national law of some countries. For instance, according to law some countries such as Austria do not allow the transfer of a citizen’s unique identifier stored on the eID to private sector applications such as public cloud service providers. Future work might include further research in being compliant with national laws to fully support an *Identity as a Service* Model. Additionally, single sign-on functionality between different cloud providers might be another future task.

References

[1] G. Kessler, "Passwords - Strengths and Weaknesses", Internet and Networking Security, Auerbach, 1997.

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

[3] J. Palfrey and U. Gasser, "Digital Identity Interoperability and eInnovation", Case Study, Berkman Publication Series, 2007.

[4] J. Goulding,: "Identity and Access Management for the Cloud: CA's strategy and vision", Whitepaper, CA Cloud Business Unit, Mai 2010.

[5] C. Emig, F. Brandt, S. Kreuzer, S. Abeck: "Identity as a Service - Towards a Service-Oriented Identity Management Architecture", Lecture Notes in Computer Science, 2007.

[6] S. A. Almulla and C. Y. Yeun, "Cloud Computing Security Management," in Engineering Systems Management and its Applications, 2010, pp. 1-7.

[7] European Commission - IDABC, "eID Interoperability for PEGS: Update of Country Profiles", 2009.

[8] H. Leitold and B. Zwattendorfer, "STORK: Architecture, Implementation and Pilots". in ISSE 2010 Conference, pp. 131-142. 2010.

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

[10] The authors are however aware that user tracking might be possible at the VIDP but we assume this entity as being trusted.

[11] http://jelastic.com

[12] https://appengine.google.com