Increasing security and privacy in user-centric Identity Management: The IdM card approach

Ronald Marx, Hervais Simo Fhom, Dirk Scheuermann, Kpatcha M. Bayarou
Fraunhofer SIT
Rheinstr. 75, 64287 Darmstadt, Germany
<last name(s)>@sit.fraunhofer.de

Alejandro Pérez
University of Murcia, Faculty of Informatics
Apartado 4021, 30001 Murcia, Spain
alex@um.es

Abstract — In this paper, we describe how security and privacy can be increased in user-centric Identity Management (IdM) by the introduction of a so-called IdM card. This IdM card securely stores and processes identity data of the card owner, an end user. The card represents a trusted device that supports the user in managing its digital identities and also in performing secure and privacy-enhanced service authentication and authorization.

Keywords—Identity Management, Security, Privacy, Smart Card.

I. INTRODUCTION

With an increasing amount of services and service providers Identity Management (IdM) is becoming crucial. Among others, IdM enables managing identity information securely in distributed or ubiquitous systems by maintaining this information at Identity Providers and by securely exchanging it to services that need for trustworthy identity information. An enormous growth is expected by [1] for the IdM market. IdMs facilitate managing information like identities, attributes, properties, and policies of its registered end users via internet. Moreover, user-centric IdMs allow users to keep some control over their personal data. That is, users control the processing of their data either directly\(^1\), indirectly\(^2\) or both.

As IdM will manage personal data of an end user and other data that is applied for authentication or authorization, the maintenance of security and privacy is vitally important for user-centric identity management. In order to assure the security and privacy across all different layers (including network or services layer) of an identity management framework, different security- and privacy-enhancing modules are needed [2]. The goal of our work presented in this paper is to introduce a so-called IdM card as one important link in this chain of modules to being able to address certain challenges in today’s IdM Systems (IdMS).

Identity fragmentation is one of the most urgent challenges that need to be addressed by IdM since it is directly related to emergence of new security and privacy risks like identity theft. Furthermore, management of several passwords and usernames often leads to password fatigue and rejection by users. Other prominent challenges and possible sources of risks like data disclosure are caused by the growing number of different entities dealing with PII or other sensitive data. This leaves the door open for illegally monitoring and tracking of data.

Another challenge related to the user-centric IdMSs is proportionality, which means the challenge to balance the traditional service providers’ wish to control their customer by means of authentication and identification, against users’ interest of being in charge of deciding who and how the user’s data is handled.

The design of an IdMS for access management while taking the risks and challenges previously mentioned is the task, where our IdM card approach makes significant contribution.

The rest of the paper is organized as follows: Section II presents related work in the area of security and privacy issues in IdM systems and introduces the SWIFT architecture as the context for our IdM card approach. In Section III we will specify requirements for the introduction of smart card in user-centric IdMS. Our IdM card approach is outlined in Section IV. This also comprises a detailed view on the contribution of our IdM card regarding the requirements from Section III. A conclusion is given in Section 0 along with open issues and outlook on future works.

II. BACKGROUND AND RELATED WORK

Recently, several approaches leveraging the potentials of electronic identity cards to implement user-centric identity management systems (especially for identifying, authenticating and authorizing users) have been proposed.

Hoepman and Kleinhuis [12] proposed an integrated architecture for access management in digital as well as physical domains, using a single smart card. The smart card is used to authenticate its holder, to grant access to physical buildings and digital applications and information. The authors claim to have implemented and introduced their integrated solution within the ICT infrastructure of a large telecommunications company (18,000 employers). This integration rests on a large scale Public Key Infrastructure (PKI) environment to support authentication, signing and encryption services for the company employees. These properties as well as the process of issuing smart cards are archived with a high level of security: The eID production takes practical constraints and users functional requirements into account, and complies with the requirements for VeriSign.
A cross-layer approach that integrates network and application layers. An IdM solution that overcomes shortcomings of existing IdM approaches for user-centric IdM exists, which leverage the potential of anonymous credential (AC) schemes. The proposals basically present several prototype implementations of an anonymous credential schema (i.e. Direct Anonymous Attestation (DAA) protocols) on a standard Java Card. The latest improvements in smart card implementation of AC have been recently achieved by Bichsel et al. and Sterckx et al. Both works proposed an efficient implementation of DAA scheme on Java Cards. While the Sterckx’s proposal still shows some of the computational limitations (due to the mathematic complexity of the DAA protocols) already identified in the early prototypes generation, Bichsel et al. claim that their solution is today’s most practicable implementation in term of computational performance (efficient proof and signature generation) and compliance to current smart card standards.

The presented mechanisms provide security and privacy preserving methods for identity management in mobile contexts. However, these mechanisms are often not feasible with respect to user privacy needs in today’s IT systems, and work only under certain assumptions, e.g. claim for practicability is made ignoring the cost for revocation operation in order to achieve acceptable computational performances.

The approach described in this paper is one of the concepts developed by the EU project SWIFT. SWIFT project develops an IdM solution that overcomes shortcomings of existing IdM solutions and addresses challenges for future IdMS. It presents a cross-layer approach that integrates network and application layer from an IdM perspective. As a consequence of a conducted gap analysis, SWIFT identified that the classical role definition of IdMS is not sufficient. SWIFT proposes a solution by subdividing the role of Identity Provider (IdP) into three different roles (cf. Figure 1) and serves as a basis for SWIFT architecture.

- Service Provider (SP), the Relying Party, consumes assertions over User’s identifying data to grant access to restricted resources, if some predefined conditions are met.
- Attribute Provider (AttP) is a sub-role of the IdP role, which exclusively focuses on the issuance of Users’ attributes, i.e. claims on the user’s identity that are certified by the AttP.
- Authentication Provider (AuthNP), another sub-role of the IdP, only assumes the responsibility of the User’s authentication.
- Identity Aggregator (IdAgg) manages Virtual Identities, which are defined as the aggregation of partial identities (i.e. set of attributes) from different AttPs. The IdAgg creates a new level on the identity management hierarchy, placing itself between the SPs and the AttPs and AuthNPs.

For the description of the SWIFT IdM architecture and its corresponding security modules, please refer to [2],[8],[9].

In user-centric IdMS, a couple of advantages are expected through integration of IdM cards. This concerns the ability to continue using services without frequent re-login as well as independence on online connections for stand-alone systems.

III. REQUIREMENTS: AN ANALYSIS FROM THE IDM CARD PERSPECTIVE

On the base of the detailed analysis of security and privacy issues in modern IdM systems, we define the following requirements for smart-card-based IdMSs. They are grouped by categories: functional, security and privacy.

A. Functional Requirements

Less dependency on online IdM entities. As entities, e.g. identity/attribute providers are legitimately considered as single points of failure that additionally may become into bottlenecks, an increased degree of independency from such components during the service (and network) access should be supported. The IdM solution should be flexible enough to allow eligible users to authenticate and access protected services or facilities, even in case of network or components failure.

Usability and mobility support. The IdM solution should be designed while keeping the end users’ convenience in mind. The end user should be able to easily perform creation, usage and deletion of her different digital identities. The support of mobility (incl. identity portability) should allow the continuity in the service usage for roaming end users moving from domain to domain or systems to systems respectively.

Scalability and openness. Provisioning of identification data to services requires a scalable storage and management of user attributes.

B. Security Requirements

High Assurance Level. Entities providing services need assurance that they offer their services only to genuine and registered users. For instance, a physical access control system need to clearly determine who is actually in front of the door, in order to apply the right admittance permissions.

Mutual Authentication. User should be able to authenticate the IdAgg and SP respectively. This requires a mutual authentication process which mitigates man-in-the-middle attacks and phishing attacks (no malicious entity should be able to impersonate a legitimate user in order to enter a facility or to access online service) especially when user’s credentials are shared among or reused by different components of the IdMS (e.g. IdAgg or SP).
Secure Storage. We stress that the relevant identity information for such a method are digital credentials, already stored on or generated by the IdM card. The access sensitive information stored on the card has to be restricted to the card owner, and the misuse of credentials (e.g., unauthorized updating of attributes) has to be prohibited.

Trustworthy Credentials. Credentials should be reliably verifiable and revocable: End users or any non-authorized entity should not be able to forge valid credentials, even if they team up. The probability for successfully forging a credential should be virtually zero. Only IdAgg (as well as IdP, AttP, and AuthNP) should be able to make genuine and verifiable assertions about a user identity. Furthermore, other security requirements should be reliably enforced, for instance message authenticity, integrity and confidentiality.

Besides the authentication of users, an IdMS must support the deployment of a fine-grained protection (in term of authorization) of physical and digital resources, as well as support for accountability.

C. Privacy requirements

The following requirements are mandatory to guarantee end-user privacy when processing (i.e. retrieval, collection, storage, disclosure, etc.) any personal data [10] in the context of smart-card-based IdMS.

Data Minimization and User Empowerment. The processing of identity data should be minimized as much as possible and only happens for legitimate purpose. Further, the solution should enable users to keep control over their personal data (e.g. through their involvement in the specification and enforcement of access rights to their private data). In order to comply with these principles, several privacy features have to be supported, which are: pseudonymity, anonymity, unlinkability and unobservability (see [11]). However, end users should not be able bypassing the non-repudiation and accountability functions while acting under pseudonyms or anonymous identifiers.

Data Privacy. The smart card-based solution should support methods to obfuscate user’s identity data before it leaves the card. This is commonly archived by means of pseudonymity and anonymity. Primarily, both concepts are useful within scenarios in which it is required to authenticate but not necessary to identify users before granting service access. Pseudonymity and anonymity alone may not be sufficient. Smart-card-based IdMSs should give the user the possibility to decide if transactions or activities are linkable or not. Moreover, the unauthorized flow of personal information should be prohibited, e.g. when interacting with untrusted terminals. The access to or disclosure of stored personal data has to be performed according to the user’s preferences.

Location Privacy. The IdMS must not collect past and current position of a user without consent.

Privacy Preservation in each IdM Workflows. Privacy should be addressed in all processes and components involved in the identity management lifecycle. Therefore, credential issuance process must guarantee data confidentiality and the underlying IdMS should support the balance between security requirements mentioned above and the users’ privacy needs.

IV. THE IDM CARD APPROACH

The introduction of the IdM card increases security and privacy in user-centric identity management. The IdM card provides a personal, secure, mobile, smart-card-like device that addresses the functional, security and privacy requirements presented in Section III.

A. Features of IdM Card

The IdM card provides a high-level interface that can be used by the end user to perform various IdM tasks. Although under personal control of the end user, the IdM card represents the IdMS and is equipped with some of its functions to enable the user to authenticate to and be authorized by services. This way, the IdM card will not be directly involved in the process of authentication and attribute provisioning, which minimizes the load and avoids the IdMS to be a single point of failure.

When looking at the typical design of IdMS, it seems to be desirable to equip the IdM card with the following functions, many of them falling into the responsibility of the IdAgg:

- PIN verification function,
- Secure storage of the user’s pseudonyms, attributes, preferences, and keys,
- Secure storage and management of the user’s IdM profile (i.e. the user’s virtual identities),
- Synchronization between IdM card and IdAgg,
- Secure storage and usage of an IdAgg signature key,
- Generation of statements.

Before access is granted, the IdM card authenticates the user by PIN. All the required information for accessing a service, is stored or generated on the IdM card and available after successful PIN authentication. The IdM card securely stores user’s attributes (verified claims about the end user like address, age, etc.), preferences (user-defined attributes), keys (for cryptographic operations), and pseudonyms (for service access). Moreover, the IdM card performs various IdM mechanisms like the storage and management of various partial identities that the user has created for different purposes. Furthermore, it is able to generate authentication and authorization statements, which are needed by the SP to allow service access to the user. In order to be accepted at the SP these have to be signed by the IdAgg. In order to sign statements on behalf of the IdAgg, the IdM card must securely store and use a signature key of the IdAgg. This key is the trust anchor of the approach and is trusted by all associated services.

The IdAgg shall employ multiple, different signature keys for the IdM cards that it issues in order to forestall the compromise of the whole system if one card has successfully been attacked. Having different keys reduces, moreover, the expected gain of an attacker. On the other hand, IdM cards shall not have individual keys to maintain unlinkability of the users. Thus, the IdM card provides a group signature that
proves the user’s affiliation to this IdAgg. Furthermore, the
user is anonymous within the group of IdM card owners with
the same IdAgg signature key.

It is very important that the storage functionality is not
static but also allows a later update initiated by the end user
and approved by the IdAgg. Attributes need to be renewed
from time to time e.g. when validation periods have expired.
Furthermore, a user may want to use new services for what he
has to include new attributes or credentials on the IdM card. In
these cases, an online update between the IdM card and the
IdAgg avoids the need to issue a complete new IdM card.

The IdM card is enrolled by an IdMS provider to its users.
After enrollment, it is under the control by the end user.
However, the user can not influence the IdM functions that are
running on the card including data like attributes and keys,
especially the IdAgg’s signature key. Thus, the card is able to
generate statements in the name of the IdMS.

B. IdM card Design

In order to be able to use the features described before, the
end user is provided with an IdM card (e.g. a smart card or java
card) where its identity data is included. We employ a Java
Card as these provide enough flexibility and storage space for
implementing the needed IdM mechanisms on the card.

The IdM card is under the control of the end user who can
use or connect it with its own device (e.g., laptop, netbook,
smart phone) or a public, stationary terminal (e.g., interactive
kiosk, digital door lock) provided that the device or terminal
supports the IdM features of the IdM card. This is achieved by
installing a client software on the terminal that handles the
user’s input and translates it into commands of the IdM card
interface (i.e. Interface: End User – IdM Card in Figure 3).

Figure 2: IdM card design

Figure 2 shows a high-level view on how the card is
structured. The Identity Manager is the main routine of the
IdM card and processes all incoming commands. Only
commands defined by the Identity Manager (i.e. for Interface:
End User – IdM Card) will be processed. Other commands or a
direct access to other components (e.g. the secure storage) are
not permitted and will be blocked. For the purpose of
generating authentication and attribute statements the Identity
Manager can consult the Statement Generation function. As the
latter relies on the user’s profile, more precisely on the attribute
and credentials from it, the Statement Generation function uses
the user_profile that is stored in the Secure Storage of the Java
Card. For this purpose, the Statement Generation function uses
the IdAgg signature key to sign the statement. This key is
securely stored in the Secure Storage and is not accessible from
the outside of the card. Also the functions implemented on the
IdM card do not allow a direct access to other components (e.g.
the secure storage) that contain attributes from the user’s profile.
Thus, the IdAgg signature key never leaves the IdM card.

An important privacy feature is the use of pseudonyms for
the communication with services. Thereby, it is necessary that
the used pseudonyms are not predictable by anyone (especially
an attacker) in order to inhibit the linkability of different
actions of sessions (i.e. service accesses). This functionality is
provided by the Secure ID Generator that generates pseudonyms if desired by the user. For the case that the user
likes to access the same service with the same pseudonym
several times (e.g. to access the same online shop account with
her shopping cart) the IdM card stores generated pseudonyms in
the Secure Storage.

If the user initiates the update function to synchronize the
user profile on the IdM card with the current version of it in the
IdAgg, this is handled by the Synchronization Client in the IdM
card. The Synchronization Client controls the synchronization
process and encrypts data to and decrypts data from the IdAgg.
The Synchronization Client communicates only with the IdAgg
that issued the IdM card to assure that no illegitimate entity
gets access to the user’s profile. Thereby, the user’s personal
data never leaves the card (or the IdAgg) without being secured
against eavesdropping and modification as the data will be
encrypted or decrypted within the IdM card.

C. Components and Interfaces

The application environment of the IdM card involves four
different components that communicate by means of the
interfaces that are depicted in Figure 3.

Figure 3: Components and interfaces diagram for the IdM card approach
1) Component: Service Provider (SP)

The SP provides some kind of service to the end user. This component receives access request from the different end users, and provides them with the service as soon as they get authenticated and authorized (as required in Section III.B). Authentication and authorization methods and their strength are service dependent. Some services only need authentication in order to prove that the end user is a real legitimate user of the service. Other services may require attribute-based authorization. The SP is urged to request just for attributes that are really necessary for providing the service. By this way, the privacy requirements are realized as no further information about the user is collected or stored during service access. The end user can also retrieve the list of required attributes (namely access policy) from SP in order to provide the proper attributes during access request.

As the SP trusts the IdAgg, it will trust the statements sent by the IdM card, which represents the IdAgg.

Network Providers (NPs) will not be considered as separate components but as special sort of SPs (for the service of network access).

2) Component: End User

On the one side, the end user represents the person who makes use of the IdM services and the SPs. On the other hand, it represents the client software on the used terminal that is acting on behalf of the user. This component has interface to the SP, the IdAgg and the IdM card. The interaction with the IdM card is performed via a client software that implements the interface of the IdM card. The terminal may be a personal mobile device or a stationary terminal that is also shared with other users. The end user has to trust the IdAgg and the IdM card for the correct processing of its identity information and for maintaining privacy.

3) Component: IdM Card

This component represents the IdM card as described in Section IV.A and IV.B.

The IdM card especially realizes the security requirement of the end user's authenticity by authenticating the end user via a PIN before access to the functions implemented on the card are granted. The card serves as a secure storage device taking care that no data is transmitted to the outside without the permission of the end user. By this way, the card supports the realization of the privacy requirements.

4) Component: Identity Aggregator (IdAgg)

In a typical IdMS the Identity Provider is responsible for the management of all identity data related to the end user. This comprises user-IDs, user-related (service specific) attributes and necessary authentication data. For attribute management, an attribute authority may work as an auxiliary instance in the background. The IdAgg’s basic functions are now taken by the IdM card, which the IdAgg has issued to the end user.

Moreover, the IdAgg complements the IdM card as a backup device for a realization of the functional requirement of service availability: In case of failure of the IdM card, the IdAgg is still able to perform the necessary authentication and authorization of the end user and to provide the necessary authentication and attribute statements. Therefore, the user can synchronize its current profile on the card with the one in the IdAgg. As stated in Section IV.B, synchronization will only take place with the IdAgg that issued the card. This addresses several security and privacy requirements.

5) Interface: End User – SP

The End User – SP interface allows the end user to request access for a service. Therefore, the end user provides the corresponding data obtained from the IdM card. If authenticated and authorized successfully, the SP provides the end user with the service.

Authentication and attribute statements are transmitted via a secure connection such that they cannot be intercepted by an attacker in order to impersonate the end user. Thereby, interface realizes the security requirements of the prevention of identity fraud and misuse of services.

The interface provides the following functions:

ServiceRequest: With this function, the end user demands its access to the service. The function accepts authentication and attribute statements as input parameters.

getAccessPolicy: This function may be used to inquire which attributes are needed by the service (i.e. which access policies needs to be satisfied) and returns a policy.

6) Interface: SP – End User

The SP – End User interface allows SP to request attributes from the end user. Therefore, SP provides its identifier and a list of attributes it expects to obtain from the end user. Similar as the interface End User – SP interface, this interface again realizes the same security requirements by a secure transfer of the attribute statements.

This interface provides one basic function:

RequestAttributes: This function carries attribute references as well as the SP’s identifier as input parameter and delivers back attribute statement (to be obtained from the IdM card).

7) Interface: End User – IdM Card

This interface is provided by the IdM card to the end user. This interface is consumed by the client software installed on the terminal used by the end user. This interface authenticates the end user before it offers the functions of the IdM card like statement generation or profile synchronization to him.

Security critical data like PIN, authentication and attribute statements are transmitted in a secure way to prevent an attacker from intercepting these data in order to impersonate the end user or misuse the services. Thus, this interface meets the security requirements as mentioned in III.B.

Following functions are provided by this interface:

Authenticate: With this function, the end user authenticates against the IdM card, using the PIN provided as parameter.

GenerateAuthSt, GenerateAttrSt: These functions are used to generate authentication respectively attribute statements on the IdM card. The SP’s identifier (to which the statement is issued) is used as input parameter and the statement is returned.
**SyncProfile**: Synchronization of the user’s profile on the IdM card and in the IdAgg is executed by this function.

8) **Interface: End User – IdAgg**

The interface provided by the IdAgg allows an end user to request attributes from the IdAgg and sync his profile on the IdM card. This interface also meets the security requirements by signing and encrypting sync packets, authentication and attribute statements.

The interface provides the following functions:

**SyncRequest**: The function takes an authentication statement and attribute references as input parameters and delivers back sync packets containing the requested identity information from the user’s profile.

**GetAuthSt, GetAttrSt**: These functions work in the same way as the GenerateAuthSt and GenerateAttrSt functions described in Interface: End User – IdM Card. It is employed if the IdAgg is utilized as backup for service authentication and authorization in case of failure of the IdM card.

V. **CONCLUSIONS AND OUTLOOK**

In this paper we presented the IdM card approach that enables a user to take its own IdM along, to make use of services independent from the utilized terminal and from the location where it is used. By this way, it furthers the mobility of the end user enabling service access from different locations, the support of multiple devices, and performing certain IdM tasks independent from the availability of the IdMS.

Regarding privacy and security, the IdM card provides the advantage that identity data is securely stored in a personal device always under the control of the user. The data terminal utilized by the user (personal mobile device or stationary terminal) is only needed for data transmission and for doing some operations exceeding the computation power of the IdM card. Additionally, the IdM card is not limited to store only one single identity but different virtual identities, which employ different pseudonyms for service access.

At present we are engaged in the implementation of a prototype of the IdM card and its application environment as proof of concept. Physical security of the smart card (for example the protection against side-channel attacks) is out of scope of this paper.

Currently, the end user has to trust the terminal with which he is accessing the card in such a way that the client software is implemented correctly and does not intercept and store the input of the user. Thus the integrity and trustworthiness of the terminal is of concern in scenarios like physical access control where the terminal is not under control of the end user. A possible enhancement for future solutions is to equip the IdM card with a biometric user authentication function instead of the currently used PIN-based authentication. The use of biometric features strengthens the binding between the IdM card and its legitimate owner. Moreover, there is no guarantee that the end user never forgets the PIN or unintentionally passes it to other users.

Standardized smart card functions only provide basic security mechanisms. Thus, as identity management becomes more important future work is the standardization of IdM card functions like the ones described in this paper.

The approach described in this paper exclusively relies on X.509 public key and attribute certificates. Other kind of certificates, like anonymous credentials as well as the related trust models could provide further interesting security and privacy properties.

VI. **ACKNOWLEDGEMENTS**

This work was funded by the EU-FP7 project SWIFT (grant number 215832). We like to thank all colleagues and project partners for supporting us in concepting, designing, and prototyping the IdM card approach.

VII. **REFERENCES**