# The SIM card as an Enabler for Security, Privacy, and Trust in Mobile Services

Carsten RUST[1], Stefano SALSANO[2], Lars SCHNAKE[1]
[1]*Sagem Orga, Heinz-Nixdorf-Ring 1, Paderborn, 33106, Germany*
*Tel: +49 5251 8891519, Fax: +49 5251 8892951, Email: carsten.rust@sagem-orga.com*
[2]*Universita' di Roma "Tor Vergata", Via del Politecnico 1, Roma, 00133, Italy*
*Tel: +39 320 4307310, Fax: +39 06 72597435, Email: stefano.salsano@uniroma2.it*

**Abstract:** The paper describes an architecture for mobile services where the SIM card is integrated for providing basic services related to security, privacy, and trust. The presented work is part of a cooperative research initiative aiming at an open architecture for mobile services. Nowadays, the security of mobile networks is mainly established through the SIM card. It provides an identity and can be used for authentication. Moreover, the SIM includes secure tamper-proof storage capabilities as well as cryptographic modules required for basic functions like signing, and ciphering. Consequently, in our architecture for mobile services, the SIM has also the role of a security token providing basic security related services. The SIM is integrated in the architecture using standard internet protocols. A web server on the card enables the exchange of data with the mobile device through HTTP. Moreover, a servlet architecture on the card allows for the provisioning of SIM services with an interface similar to that of WEB services. An important issue within the open and heterogeneous infrastructures for future mobile services is support for identification, evaluation, and rating of service offers. As an example for a SIM based service, we therefore propose a trust management service. The service is designed following the ideas of a web of trust infrastructure with an on-card key ring and trust value management. It uses digital signing for identification of services as well as for signatures by the user.

**Keywords:** Mobile services, SIM card, Trust management

## 1. Introduction

In order to repeat the success of the WEB, mobile services of the future have to be simple to find, simple to use, simple to trust, and simple to set-up. The IST project “Simple Mobile Services” [1] has the strategic objective to define a new class of services w.r.t. these requirements, meeting the specific needs of mobile users. In this paper, we mainly address the specification and implementation of mobile services that are “simple to trust”.

Typical use cases for mobile services are travel scenarios. Consider for instance a traveller arriving at an airport. Assuming that he has an electronic note (we call these notes MEMs [2]) with all his travel details on the mobile phone, he could be offered a couple of services matching his personal situation: check in for flight, guidance to the gate or other places of interest, alerts for important events like boarding, car rental for the flight destination, meetings with buddies, etc. In the described scenario, the requirement for secure and trustworthy services is obvious: for check-in, the flight ticket must be checked thoroughly. Alerts must be reliable as well as payment procedures. Finally, as service discovery and negotiation require access to personal user data, proper mechanisms for protecting the privacy of the user must be in place.

Security issues are important from both the end-user as well as from the service provider perspective. One goal for SMS is to enable individuals and small companies to become service providers (“simple to set-up”). Especially this clientele would benefit from easy-to-use and easy-to-build-in mechanisms helping them to provide secure services.

Nowadays, the SIM establishes the security of mobile networks by authenticating subscribers. Thereby, it provides an identity which is also used for further purposes as for instance micro-payment which is carried out through the mobile network operator.

As a security device for mobile networks, the SIM is also technologically capable of providing security, privacy, and trust functionality for mobile services. It includes secure tamper-proof storage capabilities as well as the required cryptographic modules. However, there is currently no established standard for communication with the SIM on application level. Hence, an application developer has no means to integrate SIM functionalities in order to secure mobile services.

In this paper, we describe our approach for integrating the SIM as an enabler for security, privacy, and trust in mobile services. The SIM can be used to provide basic security functionalities, which can – thanks to standard integration mechanisms – be utilized by application programmers when realizing mobile services. Use cases which can be handled by the SIM are, for example, secure data storage, mutual authentication between the card and the external world, signature creation and verification, data encryption, user verification, access control, or secure payment.

## 2. State-of-the-art Mobile Service Security

### *2.1 – MIDP2.0 Security Domains Concept*

A common environment for applications on mobile devices is the Mobile Information Device Profile (MIDP) for the Java 2 Micro Edition (J2ME) [3]. MIDP2.0 defines a set of APIs used by J2ME applications (MIDlets) on a mobile phone. From the security point of view these APIs are disposed in function groups. The access to these groups follows generic rules. Explicit permissions must be granted especially to APIs which are security-sensitive, create costs for the user, or provide access to the user’s private data. MIDP2.0 defines four different protection domains (Manufacturer Domain, Operator Domain, Trusted Third Party Domain, and Untrusted Domain), each granting a different level of access to the various function groups. The level of access depends on the source (origin) of a J2ME application and the level of trust this source can offer. Trust into J2ME applications is based on checking certificates which are stored either on the mobile device or on the SIM card, dependent on the protection domain of the application.

### *2.2 – SIM-based Security in Mobile Services*

SIM cards are well-established as a tamper-proof security device in mobile telecommunication scenarios. "SIM" is the abbreviation for "Subscriber Identity Module". Its primary purpose is to identify a mobile phone user to the operator's network in a secure and consistent way. In general, SIM-based applications ensure that the data used is from who it says it is from (authentication) and that it hasn't been changed without authorization (integrity) and that confidentiality is kept.

Aiming at using the SIM for basic security mechanisms, one challenge is to interface the SIM by the mobile phone. One candidate technology to this end is the “Security and Trust Services API” [4]. It offers a communication mechanism between J2ME applications (MIDlets) on the mobile phone and Java Card applications (Applets) on the (U)SIM card. The access to the SIM card itself is protected with a certificate based access control policy. This mechanism has not yet find widespread adoption by handset manufacturers. In recent years, the alterative approach to realize a web server on smart cards has been proposed as will be described later.

## 3. An Open Architecture for Simple Mobile Services

The Simple Mobile Service project has defined an open architecture for mobile services, dealing with both the execution and the creation of services. The overall architectural vision is shown in Figure 1. It includes a “Service Authoring Platform” and a “Service Execution Platform”. The architecture is open in the sense that its specifications are open and that the project is delivering an open source reference implementation.

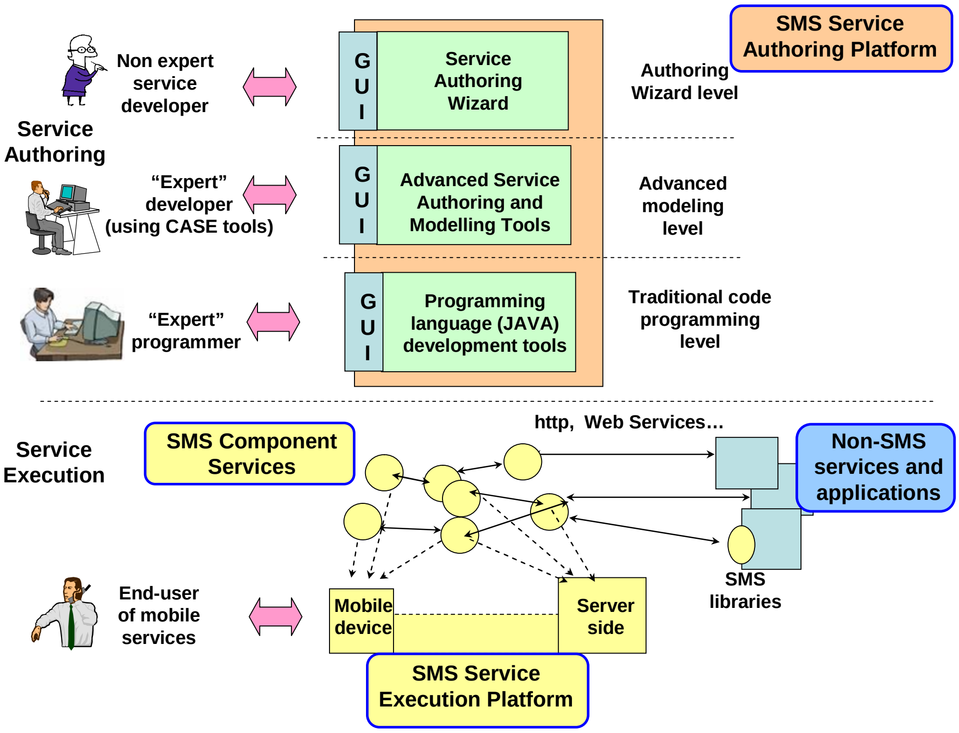

SMS Service Authoring Platform
Non expert service developer
Service Authoring
"Expert" developer (using CASE tools)
"Expert" programmer
GUI
Service Authoring Wizard
Advanced Service Authoring and Modelling Tools
Programming language (JAVA) development tools
Authoring Wizard level
Advanced modeling level
Traditional code programming level
Service Execution
SMS Component Services
http, Web Services…
Non-SMS services and applications
SMS libraries
End-user of mobile services
Mobile device
Server side
SMS Service Execution Platform

*Figure 1: Overall architectural vision*

The Service Authoring Platform is targeted to different type of developers. Non expert users use a “Simple Authoring Tool” to create and configure their services. This does not require any programming skill and it is as simple as configuring a blog on a blogger site or configuring its own home page in a web community service. On the other hand, expert programmers have access to CASE tools or to programming language APIs to compose services out of the components, exploiting the features of the Service Execution Platform.

The architecture of the SMS Service Execution Platform is based on the principles of “Service Oriented Architectures” (SOA) and it is customized to the specificity of the mobile environment. However, instead of the typically used “Web Services” paradigm, which is too resource-consuming, a lightweight middleware called “SMILE” (Simple Middleware Independent Layer) [5] is used. to the default basic communication mechanism in SMS is to use JSON [6] as serialization mechanism and to use the SIP protocol to transport the messages. The architecture is peer-to-peer rather than client-server: service components runs into the “SMS peers” and exchange information using the SMILE framework. In this context, a peer can be a mobile device but also a server offering services to other peers.

On the mobile device the Java 2 Micro Edition (J2ME) platform is used to run the “client-side” components and to realize the mobile SMS peers. The J2ME application (“Midlet”) that runs on the mobile device is called “MOVE” (Mobile Open and Very Easy).

Screenshots from this application are depicted in Figure 2 The MOVE application offers a rich set of features that can be further composed to create more advanced services. These features include:

- Handling of MEMs , which can be seen, retrieved from servers, sent to friends. used to  trigger the operation of other components etc. Users can “capture” MEMs from the environment or from other services and store them for future use.
- Outdoor as well as Indoor Navigation, used e.g. to guide a user to points of interest.
- Information services, like an information service about train schedules. These services usually are adaptations of existing WEB services to the SMS service execution platform

Note that, as shown in Figure 1 some of the above features need the support of SMS server-side components to be realized, and they may need the interaction with NON-SMS services and applications. In this last case server-side “proxy” element will typically manage the interaction with non-SMS services.

The MOVE application exploits an Open Source GUI framework named Thinlet [7] . It uses a XML based language for defining the user interface. The services can be realized by a sequence of Thinlet pages, which are handled by a “Page manager” component in MOVE. This component is like an evolved browser that manages pages by using both the traditional request/response paradigm (HTTP-like) and a “push-based” approach. In the push approach pages are originated either from trusted servers or from other trusted terminals and sent to MOVE.

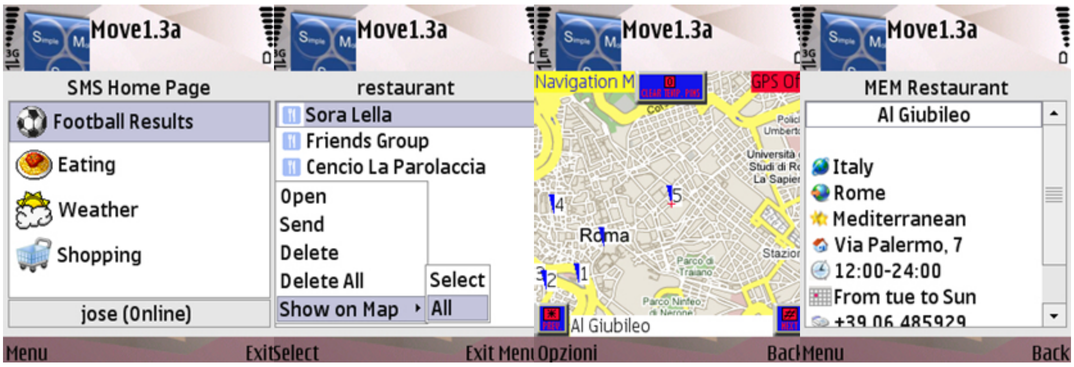


*Figure 2: Screenshots from the MOVE application*

## 4. SIM Integration

Today, access to the SIM is rigidly regulated according to specifications of large international standardization bodies. For the future however, it can be expected that mobile systems will require fast and flexible deployment for all components, as these systems will evolve and expand steadily, even after first rollout. To accommodate this requirement for the SIM, we choose the new upcoming standards for smart card implemented web services, in order to enable an open architecture and to exploit the common HTTP communication as a basis for integration into the MOVE client described above.

### *4.1 – Smart Card Web Server*

A Smart Card Web Server (SCWS) is a HTTP server that is implemented in a smart card, i.e. in the mobile context in a SIM card. The SCWS can provide static content like HTML pages or media files that hence can easily be accessed, for instance from mobile phone internet browsers.

For today’s javacard 2.2 platforms [8], the Open Mobile Alliance (OMA) has specified a SCWS [9]. It will be adapted for upcoming techniques like high speed interface or TCP enabled smart cards. To deploy advanced web services on the card, is also one of the main objectives for the upcoming Next Generation Java Card (http://www.javacardforum.org/).

*4.2 Servlet Architecture*

The Smart Card Web Server provides an extension programming interface to deploy dynamic content similar to Java Servlets from the J2EE world and other interactive modules that want to make use of HTTP communication.

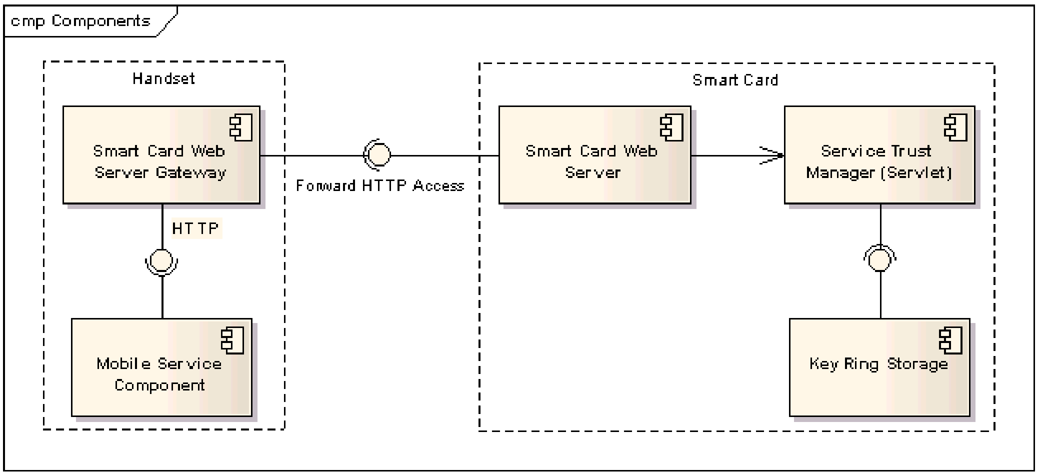
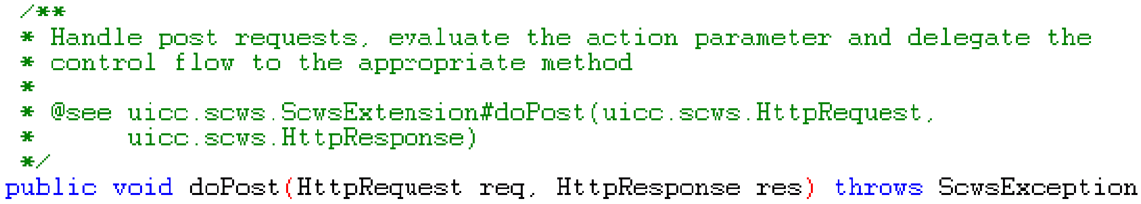


*Figure 3: Smart Card Web Server Architecture*

Figure 3 shows the basic components of the SIM service architecture to integrate the “Service Trust Manager” (cf. next section) as a Servlet of the SCWS on the SIM. Mobile Service Components can easily access the provided functions with the HTTP protocol via the “Smart Card Web Server Gateway” on a connection to the localhost address.

*4.3 – Service Integration*

For service integration, a programming interface is provided by the SCWS and card applications can implement this and register to the SCWS. The SCWS component will then delegate incoming HTTP requests to the addressed application via a callback mechanism like shown in the Java source code snipped below.



```
 /**
 * Handle post requests, evaluate the action parameter and delegate the
 * control flow to the appropriate method
 *
 * @see uicc.scws.ScwsExtension#doPost(uicc.scws.HttpRequest,
 *      uicc.scws.HttpResponse)
 */
public void doPost(HttpRequest req, HttpResponse res) throws ScwsException
```

To enable the communication, the Servlet developer can make use of the Request and Response objects. For a flexible two-way communication we mainly make use of the HTTP POST command.

## 5. SIM Based Service Trust Management

A couple of basic security services can be provided by the SIM. This includes management of sensitive user data, basic identity management, multi-id and multi-subscriber management, basic cryptographic services like signing data and verification of signatures, and many others. Based on the architecture described in the previous sections, these services can smoothly be integrated into composite services. The management of sensitive user data for instance should be only one component of the profile data management which also includes components for handling large non-sensitive data.

In the following, we present our proposal for on-card trust management in heterogeneous service infrastructures as an example for provisioning security, privacy, and trust services on the SIM.

### *5.1 – On-Card Key Management*

One of the basic concepts of our service trust management approach is that the attributes of services (including ratings resulting from previous service utilizations) are stored individually for each user. Furthermore, we propose to securely store the data on the SIM preventing manipulation of service attributes by malicious software on the mobile phone.

For identifying services, their public keys can be used, which are stored in a kind of personal keyring on the SIM as shown in Figure 4. Beyond security considerations, an advantage of this solution is portability of the stored data.

In addition to the public keys of services and of the user, also a private unique identifier (technically a private key) will be stored on the card (and never leave the card). This enables usage of the SIM as a personal security token for digital signatures and signature verification.. It assures the authenticity of the user's signature and verifies data from other parties.

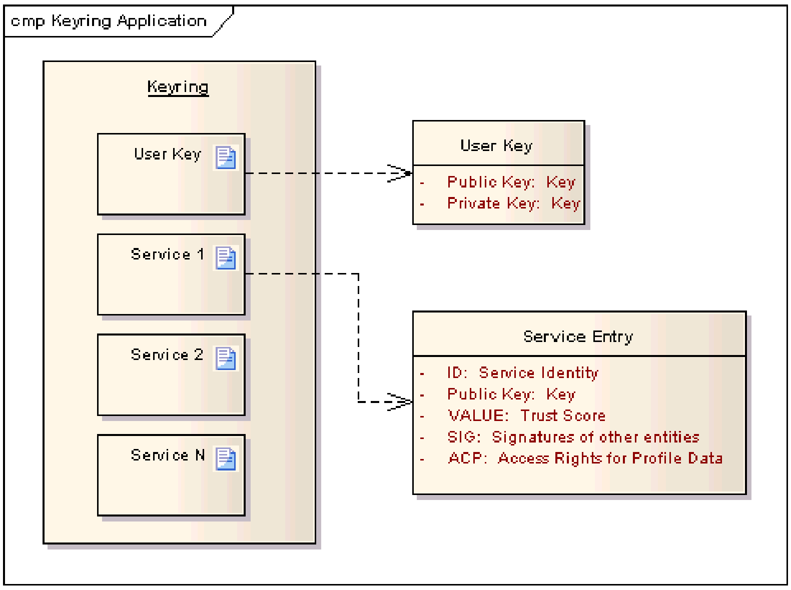


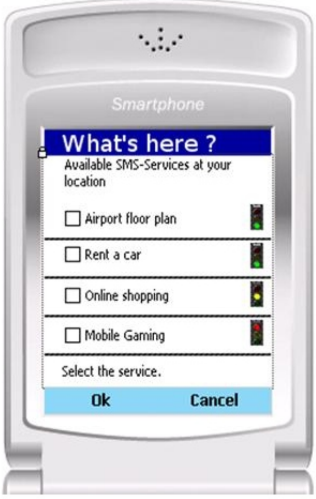
*Figure 4: Keyring on SIM card*

The Trust Management Application on the SIM card provides an interface to add and delete public keys to the internal key-ring storage and to configure their attributes. As can be seen from Figure 1, the interface component is implemented as a servlet which can be used from other applications either on the SIM or on the mobile phone.

### *5.2 – Individual Trust Level Ranking*

Every key in the key-ring application can be assigned with a discrete trust value to rank the individual service trust for the user of the service, e.g. a value between 0 and 1 as outlined in the table in Figure 5. We assume that this value will be adjusted automatically by dedicated system components, that observe each usage of a service, or based on recommendations from friends or neutral organisations. Moreover, it is conceivable that ranked public keys with trust values can be imported e.g. from an operator's database over the air or from local kiosk terminals. Finally, the user has the option to manually adust his scoring value for a service, even though most users will probably not make use of this feature.

| Trust Score | Meaning |
|---|---|
| 0 | Not trusted |
| 0.5 | Neutral trust level |
| 1.0 | Fully trusted |



What's here ?

Available SMS-Services at your location

□ Airport floor plan 🟩

□ Rent a car 🟩

□ Online shopping 🟨

□ Mobile Gaming 🟥

Select the service.

*Figure 5: Indication of trust values*

The trust score stored in the SIM might not be the only parameter to determine the trust of a service. Other, especially non static parameters like the environment could also have impact on the trust representation. The final trust recommendation is calculated during service discovery and evaluated based on user specific policies. The trust ranking of a service can be indicated to user when displaying a list of available services as shown in Figure 5. In the example the trustworthiness of each service is visualized by a small traffic light icon.

### *5.3 – Service Identity Signing*

Service entities can sign each other to prove their belonging. This can for example be used for chain stores. The signing party can also be a subordinate entity, e.g. an airport, which signs all airport related services. The signing information will also be stored in the keyring on card and can have impact on the trust level. Services unknown to the user could become more credible if they are signed by already trusted entities.

For the scenario of a traveller at the airport mentioned above, we assume that the traveller has stored the identity of the airport in his keyring. If now the so far unknown airport bookshop is being noticed, its trust level will be also identified as more trustable because its service identity is signed by the airport identity.

Hierarchical PKIs as well as Web-Of-Trust Infrastructures can become very complex. The suggested solution is to use a Web-Of-Trust, but with a limited signature chain. If we only attend direct signatures, instead of three levels like in OpenPGP [10], we are able to simplify the problem.

Inspired by [11] we introduce a trust scoring calculation based on probabilistic argumentation for mobile service authentication where users are able to gradually rate the services. Trust values of signing entities together with the trust value of the entity itself can define an average service trust, based on a simple calculation algorithm: It must be further evaluated, if e.g. neutral trusted signatures are taken into consideration.

### *5.4 Using digital signatures*

Above we described the use of digital signatures and the management of a key-ring application on a SIM to proof the identity of a service. This is important e.g. during service discovery to determine the individual trustworthiness of services.

In addition to this, we can find more aspects were digital signatures can be used: a lot of mobile services are commercial oriented and the user might want to conclude some kind of commercial contract. To secure the process, the use of digital signatures could be helpful in order to make contracts accountable. This implies an even higher level of security than the usage of secure connections which is common in B2C internet shops, as the signature outlasts the conclusion of the contract. It is important that this signature is a digital signature method that takes place on the presentation layer of the OSI model. The user must be able to see what he signs and the signature must not be lost in any transport layer or omitted in later storage.

A further use case is related to authorization. In general, authorisation is determined within an application once the identity of the other party is confirmed by authentication and the access privileges are assigned. However, the SMS concept should include another way of authorisation, allowing the user to perform certain actions or granting her certain rights that otherwise she would not have received, providing her with one-time (i.e. use only once, or time/event constrained) tokens. Several applications are possible. The user may for instance  receive a time-constrained token during check-in at the airport, allowing him to buy in tax free shops until departure. Another example is a one-time token for entering the gate.

## 6. Conclusion

Users of future mobile services need assistance to deal with the manifold service offers and options without drawbacks for security, privacy, and trust. In order to cope with this requirement, the open architecture proposed by the ‘SMS project integrates the SIM as an enabler for security related services.

During the remaining time of the project, this approach will be evaluated in demonstrators, user studies, and live trials with students at the University of Rome Tor Vergata. This evaluation will give indication to SIM manufacturers and mobile operators about the potential role of the SIM in future mobile services, in particular of user acceptance and business cases.

## References

[1] The “Simple Mobile Services” project, Project” IST 2006-034620, http://www.ist-sms.org

[2] R. Walker, G. Bartolomeo, N. Blefari-Melazzi, S. Salsano: "MEMs - Mobile Electronic Memos: efficient information capture and sharing for mobile users", Wireless World Research Forum, Meeting 18, 13-15 June 2007, Espoo, Finland.

[3] Mobile Information Device Profile 2.0, JSR 118, http://jcp.org/en/jsr/detail?id=118

[4] "Security and Trust Services API", JSR 177, http://www.jcp.org/en/jsr/detail?id= 177).

[5] S. Salsano, G. Bartolomeo, C. Trubiani, N. Blefari Melazzi: “SMILE, a Simple Middleware Independent LayEr for distributed mobile applications”, IEEE Wireless Communications and Networking Conference 2008 (IEEE WCNC 2008), March 31-April 1, 2008, Las Vegas, USA

[6] D. Crockford, JSON (JavaScript Object Notation) http://www.json.org

[7] The Thinlet project, home page, http://thinlet.sourceforge.net/home.html

[8] SUN Java Card specification V2.2.1

[9] Open Mobile Alliance (OMA) Smartcard-Web-Server V1.0

[10] RFC 2440, OpenPGP Message Format, http://www.ietf.org/rfc/rfc2440.txt

[11] R. Haenni.  Web of Trust: Applying Probabilistic Argumentation to Public-Key Cryptography. ECSQARU'03, 7th European Conference on Symbolic and Quantitative Approaches to Reasoning under Uncertainty, Aalborg, Denmark, 2003