Demonstrator on identity and trust with mobile devices

Editors:	Marc-Michael Bergfeld & Stephan Spitz (G&D)
Reviewers:	Stuart Short (SAP)
	Gregory Neven (IBM)

Identifier:	D6.2.2
Type:	Deliverable
Class:	Public
Date:	May 20, 2011

Abstract

Provides an explanation of the technology environment of the PrimeLife mobile demo and explains the technological details of the concept. Translates the findings into future privacy and identity manangement solutions in the domain of Trusted Execution Environments for Mobile Devices.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 216483 for the project PrimeLife.

Members of the PrimeLife Consortium

1.	IBM Research GmbH	IBM	Switzerland
2.	Unabhängiges Landeszentrum für Datenschutz	ULD	Germany
3.	Technische Universität Dresden	TUD	Germany
4.	Karlstads Universitet	KAU	Sweden
5.	Università degli Studi di Milano	UNIMI	Italy
6.	Johann Wolfgang Goethe – Universität Frankfurt am Main	GUF	Germany
7.	Stichting Katholieke Universiteit Brabant	TILT	Netherlands
8.	GEIE ERCIM	W3C	France
9.	Katholieke Universiteit Leuven	K.U.Leuven	Belgium
10.	Università degli Studi di Bergamo	UNIBG	Italy
11.	Giesecke & Devrient GmbH	GD	Germany
12.	Center for Usability Research & Engineering	CURE	Austria
13.	Europäisches Microsoft Innovations Center GmbH	EMIC	Germany
14.	SAP AG	SAP	Germany
15.	Brown University	UBR	USA

List of Contributors

This deliverable has been jointly authored by multiple PrimeLife partner organisations. The following list presents the contributors for the individual parts of this deliverable.

Chapter	**Author(s)**
Executive Summary	Stephan Spitz (GD), Marc-Michael Bergfeld (GD)
Chapter 1	Stephan Spitz (GD), Marc-Michael Bergfeld (GD)
Chapter 2	Stephan Spitz (GD), Walter Hinz (GD), Maximilian Loy (GD) Marc-Michael Bergfeld (GD)
Chapter 3	Stephan Spitz (GD), Walter Hinz (GD), Maximilian Loy (GD) Marc-Michael Bergfeld (GD)
Chapter 4	Stephan Spitz (GD), Marc-Michael Bergfeld (GD)

Executive Summary

This paper introduces PrimeLife's Mobile Privacy Demonstrator in the context of Activity 6 of the PrimeLife project. It is explained how the Demonstrator's technology context is dominated by SIM cards as well established identities technologies for Mobile Phones, how new Secure Elements such as SD Cards are being introduced into the market and how a future orientation towards Trusted Execution Environments can be foreseen.

Further, the G&D Mobile Security Card (MSC) / Secure Micro SD Card is introduced as the technology leveraged for the Demonstrator. It uses a Java Card Operating System and applies a Java Card Applet as "Private World" application. In this "Private World" application, protected by a "Privacy PIN" and holding the "Privacy Keys" of the exemplary eCV case, the user can interact with private data relating to his identity for a selected service, in a secure and trustworthy and encrypted manner to steer internet-based service applications which would like to have access to private data.

Briefly explained, the Demo presented here provides a privacy front-end for Mobile Devices, using the Android Operating System and a Secure micro SD Card. The Secure Element interacts with the Android OS via the "Seek" API, uses the concept of a Privacy PIN, and provides a seamless integration of front- and back-end by referring to an SMS Trigger for new Requests and communication via a secure Channel.

The lessons learned in the development of this demonstrator provides important insights for the development of now emerging technologies in the environment of Trusted Execution Environments, esp. G&D's MobiCore technology.

Contents

List of Figures

Chapter 1

Introduction: PrimeLife's Mobile Privacy Demonstrator

The demonstrator presented here is an integral part of PrimeLife's Activity 6. It has been developed by G&D in the work packages WP6.2 and WP6.3 and is closely related to the activities of the other consortium members in these work packages. In particular, the work described in D 6.2.1 (Infrastructure for trusted content) and D 6.3.1 (Advancement and integration of concepts for secure and dynamic creation of mobile services) surrounds the demonstrator presented here. Conceptually, the demonstrator is linked to the other modules of an "Infrastruture for Trusted Content" in the following manner:

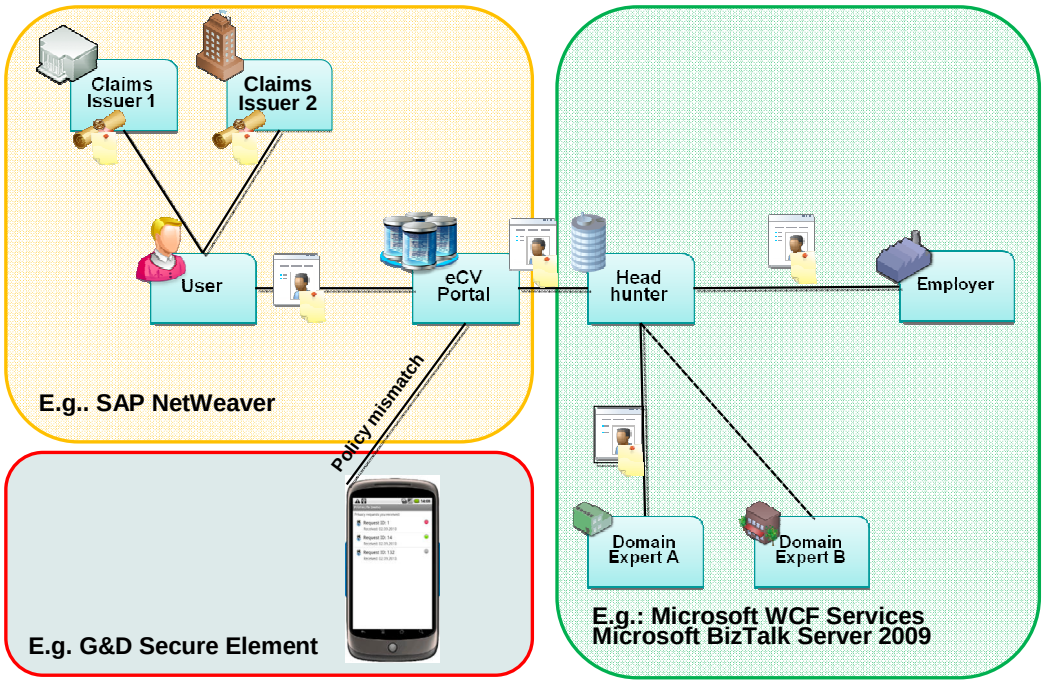

Figure 1: Mobile demonstrator in the context of A6 technologies

The backend of A6, developed largely by SAP and EMIC, and supported by the academic knowledge of Goethe University Frankfurt and the legal advice of ULD, manages an online profile of a user. In the case of the here presented demonstrator, a Job Applicant's profile is managed by an Internet-based job site. Details on how this is done can be found in particular in the Deliverables of WP 6.1 and WP 6.3.

The here presented mobile demonstrator comes into effect whenever there is a mismatch in the backend and the individual user needs to interact with his / her personal data in order to enable the backend process to continue whilst assuring privacy and security in the identity management process.

1.1 The Mobile Demonstrator's technology context and future orientation

The PrimeLife mobile demonstrator is embedded into a quickly changing market and technology environment. As shown below, the present form of providing identities and security on mobile devices is via Secure Elements such as the SIM card. The latest technology in the field of "plug-in" Secure Elements is the secure Micro SD card (µSD), which can host multiple identities and ensure secure interaction with the back end, if adapted accordingly (as in the PrimeLife mobile demonstrator).
The µSD card provides similar security as the SIM card, and more room for functions and flexibility because of a.) its capacity of storing more data and b.) its capability to manage multiple partial and isolated identities (e.g., beyond the identity provided by the network operator; these could be partial identities for loyalty schemes, payment and banking, ticketing and others) over-the-air.

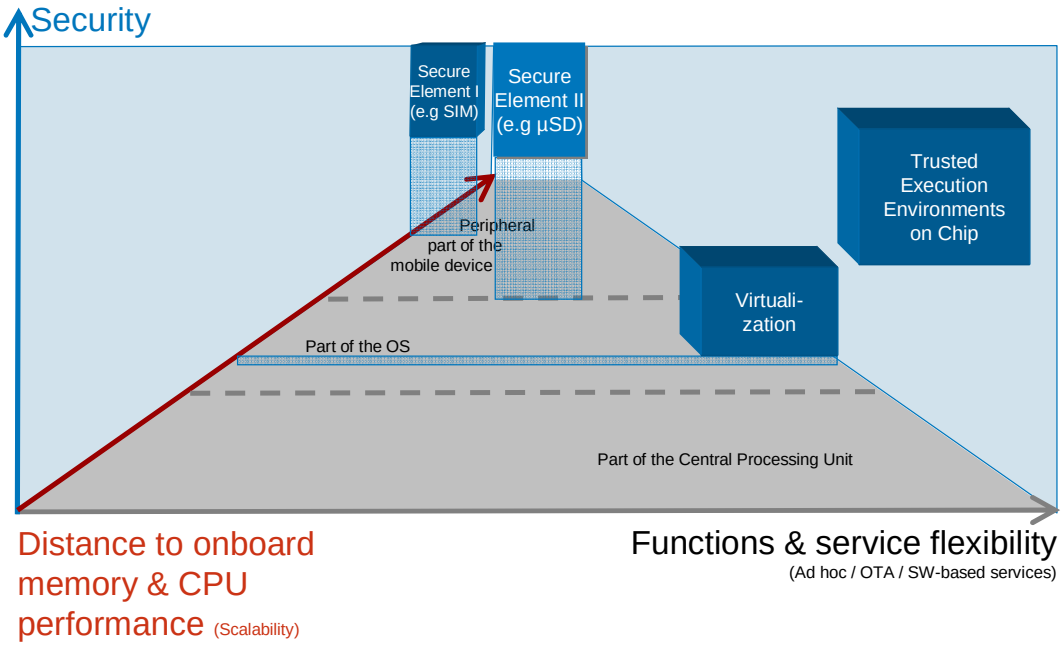

Figure 2: The technology context of the PrimeLife mobile demonstrator

For further developments in the future (also see section 4), Trusted Execution Environments will be of increasing interest due to their even further increased flexibility on their embeddedness into

the core of the chipsets for Mobile Devices. The PrimeLife demonstrator has also provided important research insights into the further development of these technologies (see section on the MobiCore technology).

In brief, the mobile demonstrator of PrimeLife pays tribute to a.) new Secure Elements for trustworthy and privacy-enhanced mobile services (e.g. by leveraging the secure µSD card) and b.) an increasingly dynamic environment of back-end / cloud-based services with which the mobile front end has to communicate in a trustworthy and privacy-enhanced manner (e.g. the back-end of the PrimeLife demo provided by SAP and EMIC).

Chapter 2

Demo description: A privacy front-end for Android

At present, the mobile ecosystem is lacking privacy functionalities and privacy-enhancing technologies are still to be rolled out on a wider scale. Different possibilities for the protection of security-critical data in conjunction with identity management infrastructures have already been identified in D 6.3.1.

Figure 3: PrimeLife demonstrator for Android

The goal of the demonstrator was to give the user a tool for managing her private identity in the cloud. It should be easy to use, meaning that it needs to be understandable from the start, without reading any further documentation.
Hence, the Mobile Device should take the role of the "Front-End" for secure and privacy-enhanced identity management. For any privacy mismatches occurring in the back-end (i.e. the Internet-based Job Portal in the eCV scenario), a request was to be sent to the mobile device, the smart phone. The user should review the mismatch and accept or decline it. The response should then be sent back to the back-end, where it is processed, in an encrypted manner.
The security of the data is maintained by a Secure Element, the MSC / Secure µSD Card, making sure that private data stays in control of the user. In order to keep the data private, even when it leaves the Secure Element, a secure channel was established between the back-end and the smart phone.

For the actual implementation of the PrimeLife demonstrator, decisions with regards to a.) the underlying Operating System, b.) the Secure Element and c.) the interfaces and d.) future orientation of technology development had to be made.

The background of these decisions is briefly explained below:

2.1 The Operating System – Android

At the time of writing, there are various mobile operating systems in the smart phone market:

- iOS (Apple)
- Android (Google, see [12])
- Blackberry (RIM)
- Symbian OS (Nokia)
- Windows for mobile phones is just being introduced

A precise look at the present growth rates of the various OSs, however, reveals the following market dynamics:

- iOS (Apple): holds 14% market share today, is expected to hold 14% in 2014.
- Android (Google): holds 17% market share today, is expected to hold 29,6% in 2014.
- Blackberry (RIM): holds 18% market share today, is expected to hold 11,7% in 2014.
- Symbian OS (Nokia): holds 41% market share today, is expected to hold 30% in 2014. With current developments regarding market acceptance of the phones and with the switch towards Windows Phone, this figure may even be revised to lower, whilst Windows Phone will claim some of the market share.

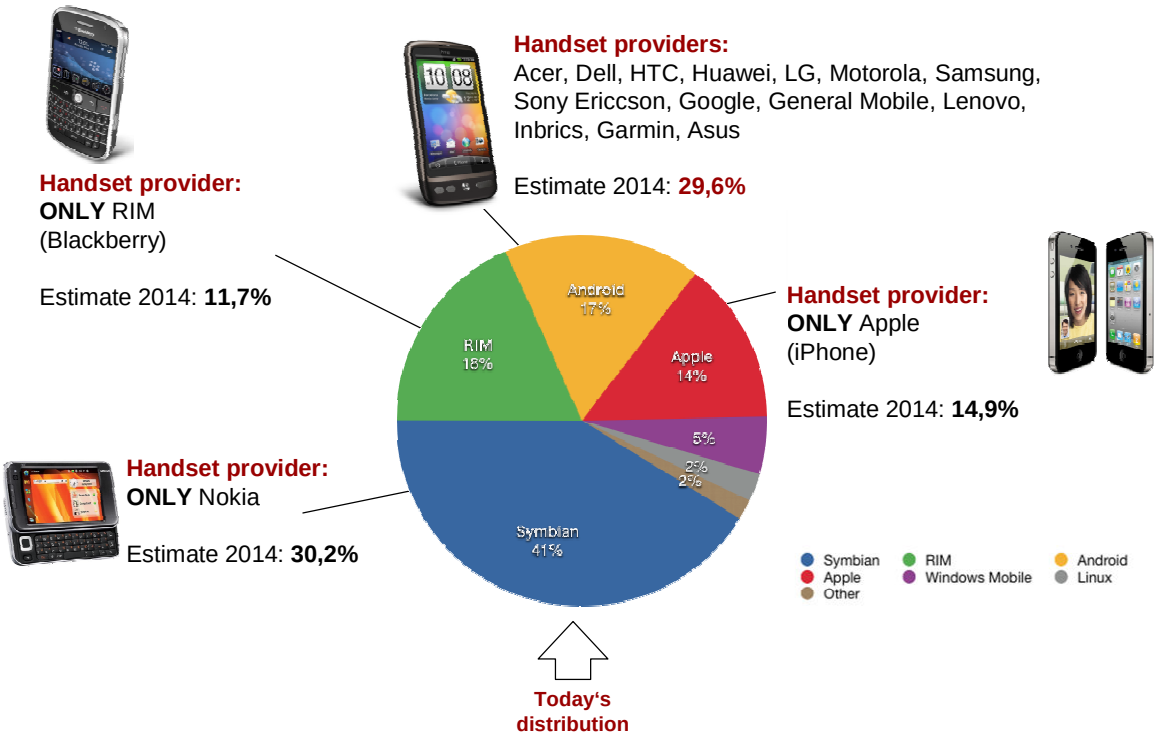

Figure 4: The technology context of the PrimeLife mobile demonstrator

Based on the above shown market situation and the forecast for 2014 (figures taken from [2] and [3]), Android (see [12]) was chosen as the Operating System on which the PrimeLife demonstrator was implemented.

At the time of development, Android was the only platform providing 3rd party Apps access to Secure Elements. This has been facilitated with the use of GD's Smart Card API for Android[1] (see section on Seek API).

This decision was not only taken due to the numbers shown above, but also due to the fact that Android has the most open system, with Google as the driving force behind it, is expected to offer very interesting, but increasingly also privacy-sensitive services. The discussion about the introduction of Google's Street View in Germany and the increasing public awareness regarding privacy and Google foster the expectation that privacy will become an important issue for the services on Android in the future.[2]

Hence, combining openness of Android with privacy protection at the same time was perceived as a very valid contribution.

2.2 Secure Storage of User Data on the G&D Mobile Security Card (MSC) / Secure Micro SD Card

For the PrimeLife mobile demonstrator, a trusted environment had to be found to hold the crucially private keys and en-/decryption capabilities to assure a trusted, secure service offering for identity management. Secure Elements (SEs) such as SIM Cards, SD Cards, or Trusted

[1] Secure Element Evaluation Kit for the Android platform http://code.google.com/p/seek-for-android/

[2] Note: The public discussion caused by the Apple case where iPhones were found to keep a life-long history of their location is another example that privacy matters in mobile phones.

Execution Environements (TEEs) can provide such an environment. In these, so called Applets (small applications) can securely be run and collaborate with the respective privacy- and identity management-enhanced services. These Applets can be installed, personalized and managed over-the-air, no matter whether they are embedded into SIM cards, SD cards or TEEs.

For the PrimeLife mobile demo the Secure Micro SD Card (see [11]) was chosen, as it combines the advantages of being:

- Capable of storing more data than a SIM card, i.e. also capable of storing multiple identities from various identity providers (the SIM-card normally only holds one identity which is the Mobile Network Operator-related one).
- Physically removable, e.g. if an individual user would change mobile phones, he could remove the Secure Micro SD card with all its identities and instantly use it in a new device.
- Structurally similar to the upcoming technologies of Trusted Execution Environments such as the MobiCore technology, so that the learnings of PrimeLife can also be leveraged for future technologies (see section on outlook and future research directions).

Technically, the G&D Mobile Security Card is a microSD card with Flash memory (2 GB) and an integrated Smart Card chip with Java Card operating system (G&D Smart Café Expert 5.0)

In the PrimeLife demo, the secure microSD was utilized

- to safely store the "Privacy Key" of the user
- to safely store the "Privacy PIN" of the user
- to be one end-point of the secure channel between the smartphone and the server.
- to provide convenient functions to encrypt/decrypt sensitive data within a secure environment.

In order to create such a secure, private and identity-related channel to the back-end, the card needs to be pre-personalized with a card specific key, namely the user's "Privacy Key". This key is derived from the service provider's individual master key using the ID of the card.[3] The (symmetric) privacy key never leaves the secure microSD as this would breach the security of the user's data.
In addition to the privacy key, the card also stores the user specific PIN, the "Privacy PIN". It authenticates the user against the secure microSD, hence needs to be entered prior to using any of the cards functionality.
Besides the "Privacy Key" and the "Privacy PIN", no other user specific data is stored on the Secure Element. All privacy requests are stored within the applications' database and encrypted via the card. Hence, if the individual user removes the card, all applications are blocked as their access to the private data is no longer given.

2.3 Java Card Operating System

The Java Card operating system, a Smart Card operating system based on state-of-the-art Java Card technology (see [4], [5], [6]), is integrated into the G&D Mobile Security Card. It supports multiple applications and complies with industry-accepted GlobalPlatform specifications.
The Java Card operating system of the G&D MSC is integrated in a highly secure Smart Card chip. It features:

[3] Note: The "Privacy Key" could also be creatd by a trusted 3rd party, being solely responsible to safeguard the privacy of the end users. For example, this could be a governmental entity.

- Secure key storage and cryptographic functionality (for example, RSA up to 2,048 bits, DES, 3-DES, AES, DSA and hash algorithms).
- Secure content protection.
- Secure memory management and encapsulated applets.
- All security-relevant methods are secured against physical attacks.

Hence, the MSC / Secure µSD Card provided a solid security technology base for the development of a mobile demonstrator with security, privacy and identity features as focus.

2.4 Java Card Operating System

A Java Card applet is a Java program that runs on the card within the Java Card Runtime Environment (JCRE, see [8]). The program and sequence logic are defined by Java Card applets which application developers create themselves. The main steps to create and install a Java Card applet include:

- Specify the functionality and the interface of the Java Card applet.
- Implement the Java Card applet as Java source code.
- Compile the Java Card applet using the standard Sun Java Compiler and the Sun Java Card Development Kit.
- Create a CAP file (=converted applet) to be loaded onto the MSC with the converter from the Sun Java Card Development Kit.
- Load and install the Java Card applet on the MSC.
- Test the Java Card applet.

This process was also followed in the development of the PrimeLife mobile demo and its Java Card applet:

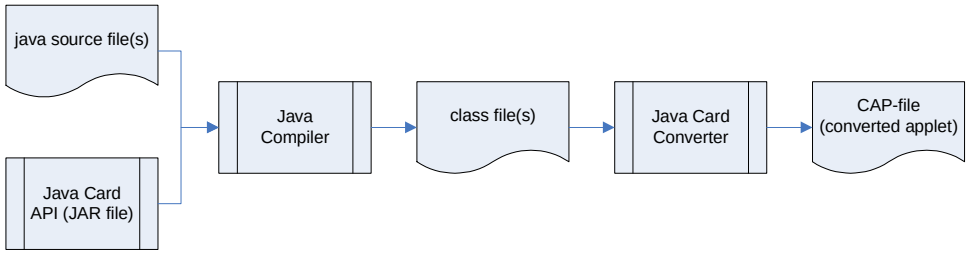

Figure 5: Development process for the PrimeLife mobile demonstrator

2.5 "Seek" – A Trusted Execution Environment API

In order to enable a collaboration between the Secure µSD card, the "Privacy Key" and "Privacy PIN" in the Applet and the Android OS of the mobile phone, an interface was needed.
The "Seek" interface (see [10]) enables the community of mobile application developers and solution providers to integrate hardware-based security features into mobile applications for

Android. Seek is a G&D technology which is increasingly[4] being made open source. Seek offers a convenient way to test and integrate Secure Elements such as the Mobile Security Card. Unfortunately, Android does not yet provide interfaces for accessing Secure Elements. Thus, the necessary software, API's and documentation are included in the Seek Developer's Kit. However, this functionality may be part of future Android versions. The supplied Smart Card API offers transparent access to Secure Elements, allowing the secure app solutions to make use of other secure form factors, such as a SIM card or an embedded solution.[5] The PrimeLife demo used "Seek" to seamlessly integrate with the Andoid system.

2.5.1 General Architecture of "Seek"

The following picture provides an overview of the architecture leveraged in the PrimeLife demo.

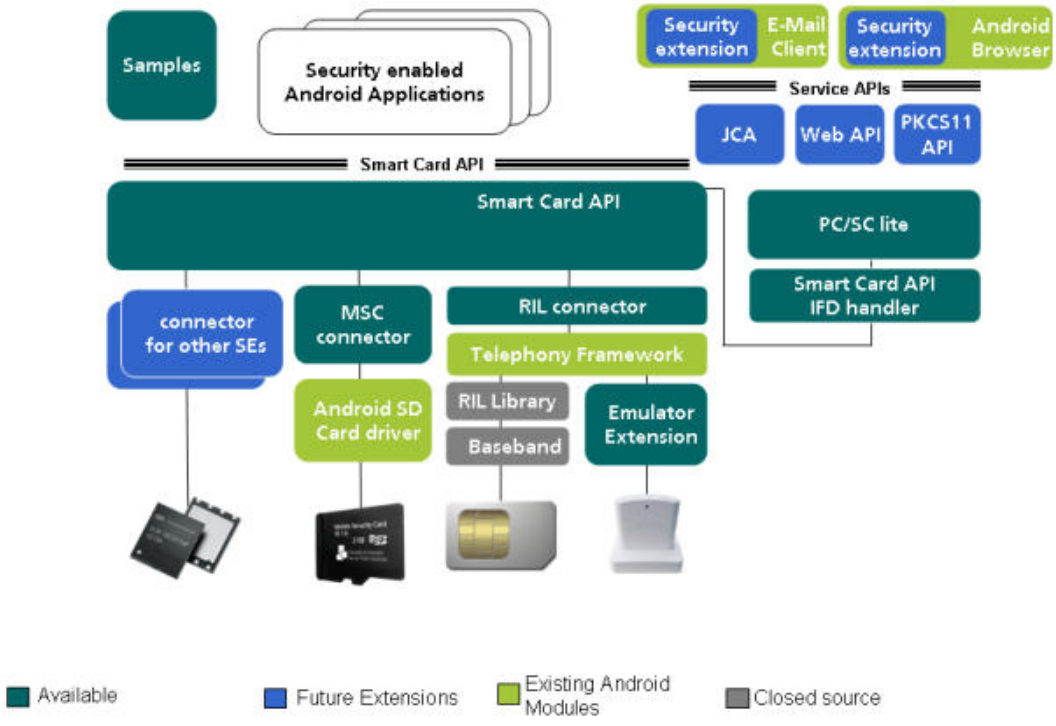

Figure 6: Overview of Modules[6]

This "Seek" architecture adds technologies known from the Smart Card domain to Android and translates the Smart Card API for the use of multiple Secure Elements such as the Secure µSD card in the PrimeLife demo. The blue software modules are included in the "Seek" Developer's Kit.
In detail, this architecture includes the following building blocks which are briefly described:

2.5.1.1 Smart Card API

[4] Note: "Increasingly" refers to the fact that the system's core and its relation to the open interfaces is evolving alongside the systems surrounding it. As the interface needs to safeguard security mechanisms, it cannot be fully open source.
[5] The Developers Kit can be obtained via https://www.cardsolutionsshop.com/shop/gi-de.
[6] Also see: http://code.google.com/p/seek-for-android/, where the interface is explained to the Google developer community.

The Smart Card API module provides a Java API for Android which allows access to various Secure Elements. The Smart Card API offers basic functionality for sending commands to a Smart Card (for example, open a logical channel and transmit application protocol data units, i.e. APDUs).

2.5.1.2 PC/SC lite

The PC/SC (short for "Personal Computer/Smart Card") is a specification for Smart Card integration into computing environments. This specification is implemented on Windows and other OSs. PC/SC lite is an open source middleware implemented for Linux (for details, see https://alioth.debian.org/projects/pcsclite/).

2.5.1.3 MSC IFD Handler

The MSC IFD Handler module connects a Mobile Security Card to the PC/SC lite system. The MSC IFD Handler is provided as binary module.

2.5.1.4 RIL IFD Handler

The RIL (Radio Interface Layer) provides a basic interface for voice, data, SMS or STK (SIM Tool Kit) functionality. In most Android phones, the RIL does not support more advanced features of current SIM implementations (for example, BIP support for OMA Smart Card web server, transparent sending of APDUs, etc.). Since RIL libraries in Android are vendor specific and closed source, the SIM in a phone can not connect via an IFD handler to PC/SC until the RIL becomes more open. However, the Android emulator provides an excellent possibility for developers to overcome this limitation.

2.5.1.5 Emulator Extensions

This module extends the Android emulator and enables it to use a real SIM in a card reader connected to the host PC. With this setup, RIL IFD handler can be used to connect SIMs to PC/SC and the Smart Card API. Other Android software incorporating a SIM can be used with the setup as well (i.e. the contact application).

2.5.1.6 Web API

This module enables the functionality of the Smart Card API in a browser and any kind of web application. Its based on Java Script (see e.g. BONDI[7] or WAC[8]).

2.5.1.7 PKCS API

PKCS API provides a higher level API for more advanced security features (for example a key/certificate store, signing, encryption, decryption, etc)[9].

---

[7] See: http://bondi.omtp.org/default.aspx

[8] See: http://www.wholesaleappcommunity.com

[9] For details, see http://rsa.com/rsalabs/node.asp?id=2124

2.5.2 Security Considerations

Android is a multi-process system, where each application (and parts of the system) runs in its own process. Most security between applications and the system is enforced at the process level via standard Linux facilities, such as user and group IDs that are assigned to applications. Additional finer-grained security features are provided via a "permission" mechanism. This mechanism enforces restrictions on the specific operations which a particular process can perform, and per-URI permissions for granting ad-hoc access to specific pieces of data.[10]

The Smart Card API leveraged in Seek and thus in the PrimeLife demo uses this permission scheme to protect access to the Smart Card. Therefore, it defines a permission "SMART CARD" which an application must request in its manifest in order to obtain access to the API. At install time, the user is asked whether or not the application should receive access to his or her Smart Card.

Access to the lower layer components such as PC/SC lite will be protected with the standard Android mechanisms.

Nevertheless, there remain weaknesses in the above described system. In particular, all interaction between the user and the µSD card still goes through the operating system on the normal touch-screen. It therefore does not protect against malware that infiltrated the operating system, or against phishing attacks by malicious apps that mimic the privacy pin dialogue. In order to prevent this, future research (e.g. in the SEPIA project) looks into empowering a dedicated hardware input interface that connects straight to the µSD card or another Secure Element, without going through the OS.

2.5.3 "Privacy PIN"

The privacy PIN, stored on the Secure µSD card (ie. the "Privacy Card"), enables the usage of the card for decryption and encryption of privacy requests.

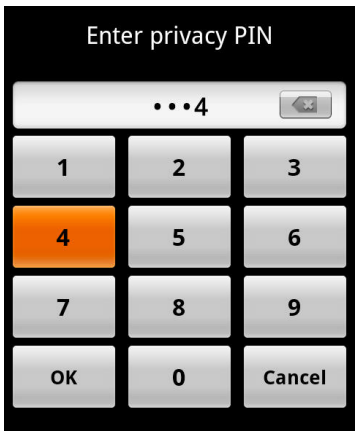

Figure 7: "Privacy PIN" entry

The "Privacy PIN", is asked for whenever the "Private World" App is started. This mechanism makes sure that, in the case of a "Privacy Card" getting lost, no other person is able to access the

[10] For more details, see http://developer.android.com/guide/topics/security/security.html.

private data. The card already comes equipped with a pre-set “Privacy PIN”, but contrary to the “Privacy key”, the user is allowed to change the “Privacy PIN”.

2.5.4 SMS Trigger for new requests

Whenever the “Private World” and the “PrivacyPIN” become relevant, the mobile front-end is triggered. In the PrimeLife demo this trigger was sent whenever a new privacy mismatch occurred in the back-end and needed approval by the individual person through the “PrivateWorld” App.
In the PrimeLife demo, the back-end sends a trigger SMS with the use of an SMS gateway. The SMS is composed of a PrimeLife tag holding a session ID (e.g. 2):

<PrimeLife>2</PrimeLife>

The “Private World” App is constantly listening to incoming SMS messages and notifies the user whenever it detects a valid PrimeLife tag within the SMS body. On the startscreen of the Andoid OS, it will show a “New privacy request available” notification, including the PrimeLife logo.

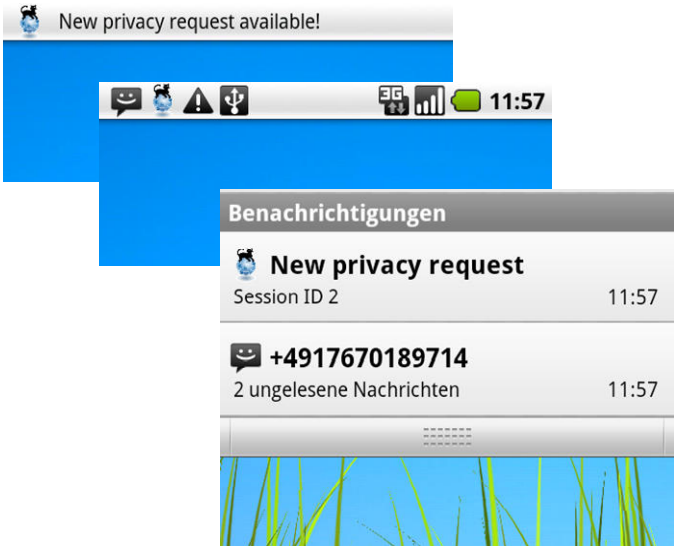

Figure 8: New request notification

Once the user selects the PrimeLife notification by tapping on the PrimeLife logo, the exact “New privacy request” will be shown. After successfully entering the “Privacy PIN” the App starts and provides the user with a list of pending requests out of which the individual user will then have to select which request to handle first.

2.5.5 Request selection

Whenever the PrimeLife App is started, the list of privacy requests is presented to the user (see Figure 9). Each request is depicted with its ID, delivery date and its state. In the case of the PrimeLife demo, there was only one possible identity provider in the back-end. If numerous services would leverage the “Private World” App, the “New privacy request” would also show which back-end service is requesting privacy authorisation.
For the state of the request, there are three possibilities:

- **Undefined**. Still unprocessed requests are in the state undefined and reflected by the grey symbol.

- **Accepted**. Any accepted request is reflected by the green symbol.
- **Declined**. Declined requests are reflected by the red symbol

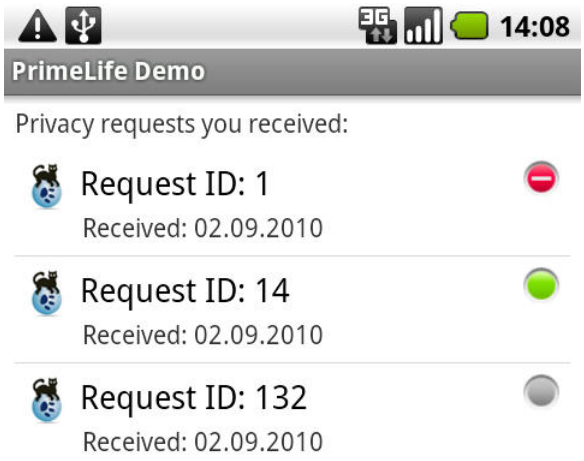

Figure 9: List of privacy requests

The user may open and interact with new requests or may delete single or all requests from the list.

2.5.6 Communication via the secure channel

Once the user of the PrimeLife demon selects one of the privacy requests from the list, the request's data gets downloaded from the back-end in an encrypted manner. Subsequently the "Private Message" is decrypted by the applet of the Secure µSD card and then displayed to the user.
Using the ID provided by the back-end via the SMS trigger, the App initiates a secure channel to the back-end (HTTP, encrypted). Therefore, the back-end needs to be a RESTful web-service supporting standard GET and POST operations for downloading privacy mismatches as well as uploading the user's response.

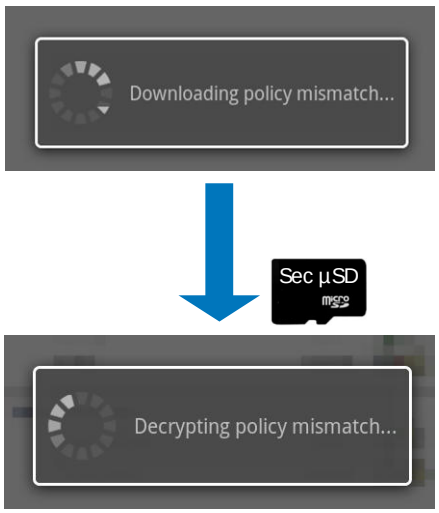

Figure 10: Downloading and decrypting privacy request / policy mismatch

2.5.7 Privacy Mismatch display

After the data package coming from the back-end has been decrypted, the XACML[11] encoded privacy mismatch gets parsed, and depending on its content various UI components are shown to the user within the mismatch view:

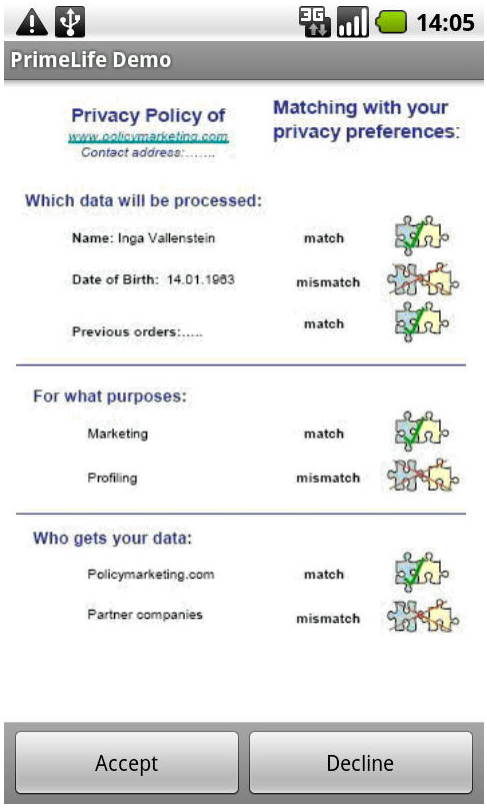

Figure 11: Mobile view of the privacy request / policy mismatch

The above shown figure is one exemplary case of how the policy mismatch / privacy request can be displayed to the end user. For example, it could be shown which data shall be processed further by the back-end. In the eCV scenario of PrimeLife, the internet-based service would, for example, want to use the full name of the person, the date of birth and previous orders. In addition, the purpose of using the data shall be profiling of the user and shall additionally be provided to partner companies. This request mismatches with the policies previously defined by the user, who, e.g. did not agree to showing the birth date, secondary usage of data and forwarding it to 3rd parties.

The user can now decide to accept the policy mismatch in this exceptional case or, alternatively decline the request.
If the request is declined, the back-end cannot process the data further. This is also the case as long as the user does not accept or decline the request. Only in the case of acceptance can the back-end server continue with its process.

[11] Note: The PrimeLife project embedded the data handling policy language in XACML, but it could in fact just as well be embedded in any XML-based language.

Once the user has made a decision on the privacy mismatch, the response is encrypted using the "Privacy Key" on the Secure µSD and then pushed back to the server.

In summary, the PrimeLife demonstrator for the mobile front-end has, together with the demonstrator developed for the back-end in Activity 6 of PrimeLife as well, provided the first ever seamless system through which back-end, internet-based processes with private data can be controlled through a front-end device in a secure and trusted manner which assures privacy and allows for the management of different identities.

Chapter 3

Demonstrator “lessons learned” for future technologies

3.1 The developments in the field of Trusted Execution Environments, esp. MobiCore technology

The lessons learned from the development of the PrimeLife demonstrator will provide important insights for the advancement of technology creation, in particular in the field of Trusted Execution Environments for mobile phones, such as the MobiCore technology.

3.1.1 Influence of the PrimeLife demo learnings on MobiCore as future technology

In order to further facilitate the required security and privacy mechanisms further that have been found and conceptualized in PrimeLife, Trusted Execution Environments (i.e., TEEs) are expected to be of particular relevance in the future. They combine increased security and additional flexibility (see figure 1) and thus trustworthy services together with cloud-based solutions to be executed even more dynamically. Herein, the TEEs can also be used as storage and processing platform for the identification of individuals and their credentials.
Both, the presented Secure µSD Card as well as the TEE by the name of MobiCore, which is currently being brought forward by Giesecke & Devrient, can be expected to be compliant to Global Platform standards and eligible for security certification. This standard has been promoted strongly by G&D throughout the PrimeLife project and first interfaces have been standardized during the course of PrimeLife (see Appendix in D. 6.3.1).
For “plug-and-play” solutions to privacy and identity management, the Secure µSD card can be used ad hoc and hence provides an important bridge technology to introduce the concept to the market place whilst TEE technologies are still under development. Most existing mobile handsets have SD card slots and could leverage the concepts developed in PrimeLife at short notice.

To take the concept of the MobiCore further, G&D is actively promoting the idea of identity isolation in standardization bodies such as Global Platform. This is a conscious step into the direction of leveraging the learning from the PrimeLife demo across the various Secure Elements in the mobile devices of the future.

For future developments with even more flexibility, G&D's MobiCore® technology can be leveraged. In order to promote privacy-enhancements and trustworthiness of mobile services in advance, the MobiCore® has already implemented the concept of isolation, which has also been leveraged in the differentiation between Private and Public World in the demo.

MobiCore is being prepared to provide a standardized and reliable solution that provides a sustainable security level across a broad range of end-consumer devices including mobile handsets, netbooks, Internet enabled Digital TVs, automotive headunits and more. It is built upon ARM® TrustZone® technology which is widely deployed by ARM silicon partners today. The MobiCore® security solution enables execution of performance-intensive and security-critical applications in a secure runtime environment, while always remaining open to new applications – via standardized interfaces. By allowing service providers to place their trust in the consumers' end devices, it enables the deployment of new high value services.
In technical terms MobiCore® strives to leverage four conceptual pillars:

- Assurance of the integrity and robustness of the whole mobile device by separating security critical processes from normal applications.
- Assurance of trustworthy user interaction between the mobile device, different Secure Elements and a service backend.
- Empowerment of numerous, separated and certified security spaces for different service and application providers on one mobile device.
- Empowerment of the integrated provisioning of various secure services to different mobile devices for independent application and service providers.

Overall, MobiCore security promotes the principle of isolation, as established in the PrimeLife demonstrator, even further. Instead of encapsulating security critical processes in extra hardware components such as, e.g.., the Secure µSD card, the main processor of the mobile device is enhanced by an additional secure execution mode – the TrustZone® Secure World. The TrustZone® Secure World resembles the concept of the "Private World" of the PrimeLife demonstrator. MobiCore is the secure operating system and is responsible for program execution in the Secure World. This will work similar to the manner in which the Secure µSD Card provided as secure execution ground for the "Privacy" application. By being built into the core chip of the mobile device, MibiCore will realize the chance to provide "Private Worlds" on mobile devices in a very cost efficient way by a combination of the ARM® TrustZone hardware and the MobiCore software.
Further, the secure runtime environment of the MobiCore (Secure / Private World) will have full access to memory and peripherals of the mobile device. Access to memory regions and peripherals defined as secure will be solely accessible by MobiCore, whereas MobiCore has access to Normal / Public World memory too.

3.1.2 MobiCore Trustlets as "Private Worlds" on Mobile Devices

MobiCore® will provide an execution environment for so called Trustlets (small applications). They are small and very specialized binaries loaded to the MobiCore runtime environment and responsible for security critical operations. Trustlets are compiled against the Trustlet API, which gives them access to secure services such as crypto-graphic functionalities, a secure keypad and methods for communication with the normal world applications. The Trustlet API can

conceptually be understood as bein similar to the above mentioned Seek API leveraged in the PrimeLife demo.
MobiCores's underlying multi-tasking kernel allows running multiple Trustlets concurrently. Trustlets run in their own processes, preventing direct memory access across Trustlet boundaries. This, again, resembles the PrimeLife demonstrator by taking the idea further that one Secure Elements such as the Secure µSD Card should be capable of processing multiple identities from different service providers.
Trustlets need to be cryptographically signed so that MobiCore can check their validity during the loading process. They can be installed, personalized and managed over-the-air. All data processed in the Secure World is strictly isolated from the data processed in the Normal World. Thus, Trustlets, are protected from data leakage or malicious intrusion via Normal World applications. Moreover, MobiCore separates the individual Trustlets from each other. This ensures that erroneous or malicious code can never influence a MobiCore®-protected Trustlet.

Hence, the MobiCore will, once development is completed, be capable of executing the use cases that have been shown in the PrimeLife demo via the Secure µSD card. The MobiCore will then be able to hold multiple "Private Worlds" with different "Privacy Keys" and "Privacy PINs" from different service providers in one mobile device and hence bring the concepts developed in PrimeLife to life in everyday products.

3.2 Next steps towards secure, private and identity providing Mobile Devices

The increasingly open operating systems cannot provide the trustworthiness that is needed to keep up with the ongoing improvements of attacks and threats, coming with the new functionalities of Mobile Devices. This means that security mechanisms need to be integrated in the system on the lowest possible level instead of being added on top. The PrimeLife demonstrator has shown how this can be implemented on a Secure Element – the Secure µ SD card. In the future, the MobiCore technology will be able to provide such a secure environment to protect security critical data and applications against attacks on software and on private data. The MobiCore will be able to encapsulate those parts which represent the essential parts and critical private data and the corresponding keys. Essentially, it will take the conceptual results of the PrimeLife demonstrator further in three steps:

1. Determine sensitive/critical information
2. Identify the parts that deal with this information
3. Isolate these parts in a Trustlet

To exemplify this with the payment Use Case, the safety critical information would be the amount of money to be transferred and the participants of the transaction (this is similar to the data processed in the eCV application of the PrimeLife demo).
Furthermore, the user has to enter PIN and TAN (i.e. Transaction Authentication Number) to identify himself to get access to the banks service and sign a transaction (this is similar to the Privacy PIN used in the PrimeLife demo).

Hence, the Mobile Devices of the future, equipped with MobiCore technology, will provide an option to translate the conceptual findings of PrimeLife's Activity 6 scenario into multiple areas of life – beyond an eCV into mobile banking, mobile ticketing, mobile social networking and the likes.

References

[1] Bergfeld, M.-M. H.; Hinz, W. and Spitz, S. (2008): Infrastructure for Trusted Content, Deliverable 6.2.1 of the PrimeLife Consortium, http://www.primelife.eu/images/stories/deliverables/d6.2.1-infrastructure_for_trusted_content-public.pdf, accessed Oct. 15th, 2010

[2] Gartner (12.08.2010): http://www.gartner.com/it/page.jsp?id=1421013

[3] dpa (10.09.2010): http://de.news.yahoo.com/26/20100910/ttc-smartphones-gartner-sieht-symbian-un-a0164be.html

[4] Sun Developer Guides, http://java.sun.com/javacard/reference/techart/intro/; http://java.sun.com/javacard/reference/techart/applet/;

[5] Sun Sun JDK: http://java.sun.com/javase/downloads/index.jsp

[6] Sun Java Card Dev. Kit: http://java.sun.com/javacard/downloads/index.jsp

[7] ISO/IEC 7816: http://www.iso.org/iso/home.html

[8] Java Card API 2.2.1: http://java.sun.com/javacard/

[9] Global Platform: http://www.globalplatform.org/

[10] Seek for Android: http://seek-for-android.googlecode.com, Global Platform: http://www.globalplatform.org/

[11] Mobile Security Card: http://www.gd-sfs.com/the-mobile-security-card

[12] Android Source: http://source.android.com