FIDIS

Future of Identity in the Information Society

Title:	"D3.1: Structured Overview on Prototypes and Concepts of Identity Management Systems"
Author:	WP3
Editors:	Matthias Bauer, Martin Meints, Marit Hansen (Unabhängiges Landeszentrum für Datenschutz, Germany)
Reviewers:	Jozef Vyskoc (VaF Bratislava, Slovakia) Sandra Steinbrecher (TU Dresden, Germany) James Backhouse (London School of Economics, UK)
Identifier:	D3.1
Type:	[Deliverable]
Version:	1.1
Date:	Thursday, 15 September 2005
Status:	[Final]
Class:	[Public]
File:	fidis-wp3-del3.1.overview_on_IMS.final.doc

Summary

The document is directed at an audience of academics, EU policy-makers, experts from technological, social science and legal disciplines and interested citizens.

It will give an overview of existing identity management systems (IMS). Different types, classes and subclasses of IMS are identified, described and illustrated by examples of existing IMS. To get an overview of the variety of existing technical implementations different designs of IMS are presented. Privacy enhancing mechanisms are developed and selected corresponding privacy enhancing technologies (PET) are shown as examples of existing implementations of those mechanisms. Finally an overview is presented of current research and development activities on IMS and conclusions, especially from the FIDIS Network of Excellence.

The FIDIS NoE receives funding from the Community's Sixth Framework Program

Copyright Notice:

PLEASE NOTE: This document may change without notice – Updated versions of this document can be found at the FIDIS NoE website at www.fidis.net.

Members of the FIDIS consortium

1. Goethe University Frankfurt	Germany
2. Joint Research Centre (JRC)	Spain
3. Vrije Universiteit Brussel	Belgium
4. Unabhängiges Landeszentrum für Datenschutz	Germany
5. Institut Europeen D'Administration Des Affaires (INSEAD)	France
6. University of Reading	United Kingdom
7. Katholieke Universiteit Leuven	Belgium
8. Tilburg University	Netherlands
9. Karlstads University	Sweden
10. Technische Universität Berlin	Germany
11. Technische Universität Dresden	Germany
12. Albert-Ludwig-University Freiburg	Germany
13. Masarykova universita v Brne	Czech Republic
14. VaF Bratislava	Slovakia
15. London School of Economics and Political Science	United Kingdom
16. Budapest University of Technology and Economics (ISTRI)	Hungary
17. IBM Research GmbH	Switzerland
18. Institut de recherche criminelle de la Gendarmerie Nationale	France
19. Netherlands Forensic Institute	Netherlands
20. Virtual Identity and Privacy Research Center	Switzerland
21. Europäisches Microsoft Innovations Center GmbH	Germany
22. Institute of Communication and Computer Systems (ICCS)	Greece
23. AXSionics AG	Switzerland
24. SIRRIX AG Security Technologies	Germany

Versions

Version	*Date*	*Description (Editor)*
0.1	31.01.2005	Structure of the document and first draft set up, Contribution from ALU-FR integrated (Chapter 5.4)
0.2	10.02.2005	Contribution from KU Leuven and IBM ZRL (Chapter 5.5), TUB (Chapter 5.1) further work of ICPP on the Chapters 2,3,5, 6 and 7, 8, 9 and 11
0.3	16.02.2005	Contribution from ISTRI integrated (Chapter 5.3), additions in Chapters 2, 3, 6 and 7 by ICPP
0.4	28.02.2005	Contribution from EMIC integrated (Chapter 5.2), further examples and comments in Chapter 4.4.1 added, Chapter 1 added
0.5	24.03.2005	Remarks from the reviewers integrated
1.0	31.03.2005	Final version put together
1.1	15.09.2005	Integration of the remarks of the external reviewers

Foreword

FIDIS partners from various disciplines have contributed as authors to this document. The following list names the main contributors for the chapters of this document:

Chapter	*Contributor(s)*
1	Martin Meints (ICPP)
2	Martin Meints (ICPP)
3	Marit Hansen, Martin Meints (ICPP)
4	Ioannis Maghiros, Elsa Lignos, Sabine Delaitre (JRC), Marit Hansen, Christian Krause, Henry Krasemann, Martin Meints (ICPP)
5	Frank Steuer (TUB), Gábor Hontert (ISTRI), Christian Geuer-Pollmann (EMIC), Sven Wohlgemuth (ALU-FR), Claudia Diaz (KULeuven), Michael Backes (IMB ZRL)
6	Marit Hansen, Henry Krasemann, Christian Krause, Matthias Bauer, Martin Meints (ICPP)
7	Matthias Bauer, Martin Meints, Marit Hansen (ICPP)
8	Martin Meints (ICPP)

Table of Contents

1 Executive Summary ... 8

2 Introduction ... 11

2.1 Scope of this Document ... 11
2.2 Structure and Content of this Document ... 11

3 Definition of Types of IMS ... 13

4 Structuring IMS ... 15

4.1 Introduction ... 15
4.1.1 Developed Structure for the Database on IMS ... 15
4.1.2 Product Oriented Classification of IMS ... 15
4.2 Structure for the Database ... 16
4.3 Structuring Identity Management Systems from the Product View ... 19
4.3.1 Class 1 IMS ... 19
4.3.2 Class 2 IMS ... 23
4.3.3 Class 3 IMS ... 24

5 Designs of IMS ... 25

5.1 Liberty Alliance and Sun Java Access Manager ... 25
5.1.1 Liberty Alliance ... 25
5.1.2 Sun Java System Access Manager ... 26
5.1.3 Summary ... 31
5.2 Web Services, the Claims-Based Security Model and Federated Identity Management ... 32
5.2.1 Introduction ... 32
5.2.2 WS-Security ... 33
5.2.3 WS-Trust ... 34
5.2.4 WS-Federation ... 34
5.2.5 Summary ... 36
5.3 Case Study: Enterprise Identity Management in a Bank ... 37
5.3.1 Premises ... 37
5.3.2 Feasibility ... 38
5.3.3 Pilot Project ... 41
5.3.4 Administration Related to the Introduction of the EIM System ... 47
5.3.5 Long Term Functional Expectations, Future Prospects ... 49
5.3.6 Summary and Assessment ... 51
5.4 iManager – Identity Manager for Partial Identities ... 52
5.4.1 Identity Management with Partial Identities ... 52
5.4.2 Data Structure of a Partial Identity ... 53
5.4.3 Architecture of the iManager ... 54
5.4.4 Summary ... 56
5.5 The idemix Credential System ... 57
5.5.1 Basic Credential Protocols ... 57
5.5.2 Credential Options and Attributes ... 59
5.5.3 Parameters of the Show Protocol ... 59

5.5.4 Summary ... 61

6 Privacy Enhancing Concepts ... 62

6.1 Mechanisms to Meet Requirements of IMS with Respect to Privacy ... 62

6.1.1 Introduction ... 62

6.1.2 Type 1 IMS: Account Management Systems ... 63

6.1.3 Type 2 IMS: Profiling ... 63

6.1.4 Type 3 IMS: User-Controlled Context-Related Role and Pseudonym Management ... 64

6.2 Good Practice Examples and Considerations of Privacy Enhancing Technical Implementation of These Mechanisms ... 66

6.2.1 Type 1 IMS: Account Management Systems ... 66

6.2.2 Type 2 IMS: Profiling Systems ... 66

6.2.3 Type 3 IMS: User-Controlled Context-Dependent Role and Pseudonym Management ... 67

6.3 Summary ... 69

7 Research and Development Topics ... 71

7.1 Research and Development in Type 1 IMS ... 71

7.2 Research and Development in Type 3 IMS ... 71

7.2.1 Identity Commons / OASIS XDI/XRI Standards ... 72

7.2.2 LID - Light-Weight Identity Management ... 72

7.2.3 Biometric Secured Client-Side Identity Management ... 72

7.2.4 Integration, Advanced Prototyping and Basic Research ... 73

7.3 Summary ... 73

8 Conclusion and Outlook ... 74

9 Glossary ... 76

10 References ... 78

11 Indices ... 81

11.1 Index of Figures ... 81

11.2 Index of Tables ... 81

1 Executive Summary

This document gives an overview on existing identity management systems (IMS). Using definitions established in the FIDIS Network of excellence (Deliverable 2.1) taking a look at the procedures of the management and the data managed leads to **three types of IMS**:

1. *Type 1: IMS for account management,*
   implementing authentication, authorisation, and accounting,
2. *Type 2: IMS for profiling of user data by an organisation,*
   e.g. detailed log files or data warehouses which support e.g., personalised services or the analysis of customer behaviour,
3. *Type 3: IMS for user-controlled context-dependent role and pseudonym management.*

A search on existing implementations of IMS including prototypes and concepts leads to **three classes** of solutions:

1. Class 1: Pure IMS whose main objective is to support or implement identity management functionality
2. Class 2: Systems/applications with another core functionality, but based on and thereby supporting at least some identity management functionality
3. Class 3: Systems/applications which are independent from identity management functionality, but nevertheless offer at least some identity management functionality, such as add-ons

In this document 60 IMS were investigated and categorised in these three classes and 14 corresponding purpose oriented subclasses. Most of the examined type 3 IMS are tools and programs with partial functionality of IMS; they are not integrated solutions.

To get an overview of the variety of existing technical implementations different **designs of IMS** are presented. These examples are focused on IMS of type 1 and 3; IMS of type 2 will be covered in Deliverable 7.2 within the FIDIS NoE. In this chapter relevant standards and basic technologies such as “Liberty Alliance”, XML/SOAP and the “*idemix*” credential system are presented. In addition, examples of existing implementations of IMS (Sun Java Access Manager as type 1 and *iManager* as type 3) showing the main functionalities and the basic architecture of such systems are discussed. Finally, an example of good practice for the implementation of an IMS (in this case type 1) is presented, i.e. a project in a Hungarian bank including the analysis of the requirements and the selection and implementation of an appropriate technical solution.

An additional important part of this document is the description of **mechanisms with respect to privacy** for IMS. Ten main mechanisms related to the main functionality of IMS: security, privacy enhancing technologies and designs, interoperability and a successful market penetration are introduced and discussed in context of the three types of IMS. For type 1 and type 3 IMS, recommendations and examples of technical implementations of these mechanisms are presented. This structure can be used to categorise existing **privacy enhancing technologies (PET)** for IMS.

According to studies carried out e.g. by the Gartner Group, the Yankee Group and the Radicati Group, the **market of IMS of type 1** is expected to grow fast at least until 2008. Turnaround prognosis starts from 748 million US $ in 2004 and varies from 3 billion US $ in 2007 to 10.2 billion US $ in 2008. Technologically we observe a trend of further integration of related solutions such as customer relationship management systems (CRM) and further decentralisation of the account administration (Federated IMS).

For type 3 IMS we observe a **technological trend** towards new standards such as OASIS XDI/XR.

In general we notice that the originally quite strict borders **between the defined three types of IMS are diminishing**. Type 1 IMS (account management systems) currently are expanding towards customer relationship management (CRM), which could as well be used in the context of type 2 IMS. In addition to the organisation-side view, type 2 IMS (profiling systems) have a client-side view, which could as well be considered to be identity management of type 3. The categorisation into three types originally designed for different products still serves well to describe a certain view on more and more integrated solutions.

Using the economic lifecycle model for products with the identified types of IMS, we observe that IMS of the types 1 and 2 are in the second phase (expansion) of this model. The mechanisms of market (such as the competition between various manufacturers, supported standards like XML/SOAP, LDAP, SQL etc.) are working quite well with these types of IMS. Looking at IMS of type 3, we observe that they are in the phase “experimental” of the economic lifecycle model. The large variety of existing solutions presented in this document, the low degree of commercial activities (compared to the IMS of Type 1 and 2) and significant public activities (public promoted projects, public research) lend support to this classification.

Looking at technological aspects of the described types of IMS, there is no public technology promotion necessary for IMS of type 1 and 2. Areas of research and development are integration of related and so far independent systems and technologies. This could lead to further development of the framework of European legislation (especially in the sector of privacy compliance) or its application.

While the necessity for activities in the legislation is the same with **type 3 IMS** as with IMS of type 1 and 2 **there are additional needs**. Barriers towards expanding markets and possible activities for overcoming those barriers are:

- The perception differs widely of what identity management is. A clearer taxonomy and public awareness are necessary.
- While current concepts and technologies for identity management are not commonly understood, new technologies such as RFID and Ambient Intelligence are emerging. The technical opportunity of remote readout of e.g. the RFID without any notice by the user raises new questions towards identity management. Most today established IMS know an authentication done actively by the user.
  In addition, known technologies such as the use of mobile devices and biometrics are developing towards new services or applications (e.g. location based services and ID documents). The public reception influenced by technology friendly placement and a lack of integrated concepts is dominated by the discussion of risks. Technological, political, social and economic opportunities have to be looked at in combination with legislation (including human rights and privacy compliance). As a result there will be

recommendations for further integrated technological development and development of legislation towards those technologies.

- Integration of the existing, technologically feasible solutions is generally poor, interoperability therefore a major area of interest.
- While there are some prototypes with good usability features (e.g. *iManager*), many tools and application examined in this document are of poor usability (e.g. first generation remailer). This applies especially to those tools addressing special technical solutions for privacy. To gain a better acceptance in the market usability has to be improved.
- For type 3 IMS privacy, compliance is a unique selling proposition. On the other hand dependability and risk minimisation (understood as elements of security) are important for the provider of commercial or governmental services. This disjunction is leading to a separate discussion on fraudulent use together with criminal and forensic aspects of identity and identity management. Recommendations for further development of legislation based on an integrated understanding of the underlying technologies and social systems could be one result of this discussion.

2 Introduction

2.1 Scope of this Document

This document is directed to an audience of academics, EU policy-makers, experts from technological, social science and legal disciplines and interested citizens. It contains an overview on existing and planned identity management systems (IMS). The overview is by no means comprehensive, but major types, classes and subclasses of IMS are described and examples of existing IMS including some prototypes and concepts are given. In addition representative designs are included as privacy enhancing concepts illustrated by examples of implementation of those concepts.

This study is focused on IMS types, such as account management systems (type 1) and user-controlled context-dependent role and pseudonym management (type 3). For the IMS type profiling systems (type 2), only a few examples are given. Profiling systems will be examined under technological, privacy-related and legislative aspects, among others, in Workpackage 7 (Deliverables 7.2, 7.3 and 7.4).

2.2 Structure and Content of this Document

This document is divided into four parts:

Part 1: Definition of types and structure of IMS

Part 2: Examples of good design of IMS

Part 3: Privacy enhancing concepts including good practice examples

Part 4: Research and development in the area of IMS and conclusions.

In the **first part** we define three types, three classes and various subclasses of identity management systems (IMS) (Chapters 3 and 4). A commented list on IMS gives an overview on existing implementations.

In the **second part** examples of typical designs for type 1 and type 3 IMS both from market and research labs are described (Chapter 5). This starts with the *Liberty Alliance* and the *SUN Java System Access Manager* (both IMS type 1) followed by an introduction in the concepts of claims-based security model and federated identity management (type 1) for web-services The examination of type 1 IMS is concluded by a case study of the introduction of an IMS type 1 in a bank in Hungaria. This is followed by the *iManager* (IMS type 3), a prototype of an identity management system developed for personal digital assistants (PDAs) gives an impression, how location based services can be introduced in IMS. The chapter is concluded by the presentation of *idemix*, a credential and zero knowledge protocol based identity management system (type 3). Type 2 IMS (profiling systems) will be discussed in the Deliverable 7.2 and not in this document.

The **third part** lists privacy enhancing criteria for all types of IMS. This part is concluded by considerations and good practice examples of the implementation of the described mechanisms (Chapter 6).

The **fourth part** looks into current areas of research and development of IMS. Relevant findings are summarised and conclusions especially for the FIDIS Network of Excellence are presented (Chapters 7 and 8).

3 Definition of Types of IMS

Using the understanding and the definitions established within the FIDIS Network (Workpackage 2, Deliverable 2.1 "Inventory of topics and clusters"[1] and Wiki definitions[2]), we understand Identity Management as the management of digital identities or digital identity data. There are several approaches which differ, e.g.:

- In the procedure of management (by whom? which operations on data possible?)
- In the type of managed data (personal or organisational data? comprehensive profiles or selection of roles or partial identities? privacy or identifiability?).

Taking a look at the market of existing IMS, on prototypes, concepts and IM-related tools we observe three main types of IMS:

1. Type 1: IMS for account management,
   implementing authentication, authorisation, and accounting[3],
2. Type 2: IMS for profiling of user data by an organisation,
   e.g. detailed log files or data warehouses which support e.g., personalised services or the analysis of customer behaviour[4],
3. Type 3: IMS for user-controlled context-dependent role and pseudonym management [ICP03].

Identity management systems of type 1 and 2 are mainly used by organisations (institutions, enterprises etc.), especially bigger ones. The approach to use and to manage them is basically a centralised; administration usually is done by selected administrators or operators and not by the user her- or himself. As a result, we find mainly commercial implementations of those types of IMS. The data managed are personal as well as organisational, depending on the environment and purpose in or for which the IMS is used. Reliable identification of persons or reliable assignment of the profile to a person is usually the main focus of those systems, not privacy.

Type 1 IMS were originally defined as account management systems, used within an organisation especially for account and access administration for computers and network services (e.g. the Windows-NT-Domain-concept by Microsoft, NIS by SUN etc.). Today directory services are used, storing personal data for extended use, e.g. in the environment of human resource management (e.g. Microsoft Active Directory together with Microsoft Exchange and SAP HR).

[1] See http://www.fidis.net/293.0.html

[2] http://internal.fidis.net/178.0.html?&no_cache=1&tx_drwiki_pi1[keyword]=t2.1%20definition

[3] http://infosecuritymag.techtarget.com/2002/apr/cover_casestudy.shtml,
http://www.oracle.com/technology/products/id_mgmt/index.html
http://www3.ca.com/Solutions/ProductFamily.asp?ID=4839

[4] http://www.lumeria.com/what.shtml
http://www.epic.ca/TechnologyDay/October05_2004/
MoreInformation/Presentations/RandallBartsch%20-%20Identity%20Mgmt.pdf

Type 2 IMS will be the subject of Deliverable 7.2 (Inventory on actual profiling techniques and practices) and will therefore not being discussed in depth in this document.

Type 3 IMS are characterised by the user control as basically decentralised, user and client-orientated (Management by whom? management done by the user). The data managed are mainly personal data. Privacy protection therefore is a driving force for the development of IMS of this type and a relevant unique selling proposition (USP). To implement certain functions, such as use of trusted pseudonyms or authentication (e.g. via credentials), in some cases the implementation of centralised third party services is necessary. In addition the communication partner of the user, who is contacted via the managed identity, in many cases is an organisation.

Examining more closely the market of type 3 IMS, we find many partial solutions. They are mainly client-side tools and applications. We find them mostly developed outside the commercial sector (open source, freeware, research and development within public projects and universities).

4 Structuring IMS

4.1 Introduction

4.1.1 Developed Structure for the Database on IMS

The following structure is derived from “Identity Management Systems (IMS): Identification and Comparison Study” [ICP03]. Starting with the listed information about implementations of IMS in the study, a database structure was developed and discussed to meet two requirements within the FIDIS Network of Excellence (NoE):

1. to serve as a basic structure for the database in WP 8.3 (database on IMS)
2. to serve as guideline for the contributions of the FIDIS-partners in Chapter 5 of this deliverable.

Within the development of this structure, many additional data fields with additional information were discussed, e.g., specific information on identity management, password management and various other mechanisms listed in Chapter 7 of this deliverable.

To get a broader, less deep overview of IMS, these approaches will not be described in the initial version of this deliverable and the first version of the database. After the completion of this work the information available on IMS will be again discussed from the basis of the ideas mentioned above. The results of the final discussion upon the structure and the proposed steps of later development of the structure of the database will be documented within D8.3.

4.1.2 Product Oriented Classification of IMS

Using a more product oriented view we developed a new classification on IMS of the described three types (see Chapter 3 of this deliverable). This view is motivated by the following perception that in general there are three main classes of IMS:

1. Class 1: Pure IMS which main objective is to support or implement identity management functionality
2. Class 2: Systems/applications with another core functionality, but basing on and thereby supporting at least some identity management functionality
3. Class 3: Systems/applications which are independent from identity management functionality, but nevertheless offer at least some identity management functionality as add-on

Additionally there are systems/applications related to IMS that prepare the ground for identity management, e.g., anonymising services or tools for matching privacy preferences as well as some third party services. As they represent important concepts tightly connected to identity management functionality, they will also be briefly mentioned in this document.

Within those classes, subclasses can further be distinguished which focus on specific identity management functionalities meeting the requirements listed in Chapter 6 of this deliverable. They will be listed in Chapter 4.3 and examples of existing IMS for the introduced subclasses will be given.

4.2 Structure for the Database

Attribute Label	Definition	Values
Name	Name of the IMS	Text
Version number	Version of IMS	Text
Manufacturer / distributor	Main manufacturer or provider of the IMS	Text
Nature of provider / distributor	Description of the nature of the provider of the system. (e.g. public, private, regional, national, international)	Text
Nation	Nation of the manufacturer's resp. of provider's location	Text
Geographical scope		Enumerated: National, European, International
Supported languages		Text
State of development	Statement whether the IMS is an available product or a service on the market (Available), still a Prototype, a Suspended prototype, or just a Concept.	Enumerated: Available, Prototype, Suspended prototype, Concept
Type of IMS	1: Access Management System; 2: Profiling System; 3: IMS for user-controlled context-dependent role and pseudonym management	Enumerated: see left
Class of IMS	Class 1, 2 or 3 as per definition of types outlined in section 1	Enumerated: type 1, 2 or 3
Closed/open IMS	"Closed IMS" means that the scope of the managed identities is restricted to the IMS context. "Open IMS" means that the managed identities work with several systems or applications.	Enumerated: Closed, Open

Attribute Label	Definition	Values
Main functionality	Short description (e.g. form fill-in, single sign-on, etc.)	
Technical requirement	Description of the hardware, software, operating system and services the IMS requires	Text
Price	Price of the IMS	Integer / Text
Installation base of the IMS	Number of users of the IMS / Penetration of market	Integer / Text
Interoperability / standards	Description if the IMS can be used with other applications and systems. This could be achieved by using standards like protocols for communication.	Text
Seals etc.	Description of seals and other awards, etc.	Text
Server-side component(s)	Description of the server-side-components (data storage and processing)	Text
Control of the server-side data by the user	Description of methods (e.g. encryption), how the control of the user over his identity related data is established with respect to availability, integrity and confidentiality	Text
Client-side component(s)	Description of the client-side-components (data storage and processing)	Text
Support by third parties	Description of which third party or intermediary support is integrated, e.g., certificate providers, IMS providers, delivery services, payment services, ...	Boolean (yes/no) + text
References	Sources of information, literature, links etc.	Text

Attribute Label	Definition	Values
Description of functionality / features (client and server)	What are the characteristics / specialities? Handling of identities? Use of pseudonyms / roles? Support of anonymity? Use of electronic signatures / PKI? Storage of data? Handling of accounts? Password management? Security / encryption, etc.? Data protection? Privacy Enhancing Technologies? Data minimisation? Support of law enforcement? Usability? […]	Long free text
Screenshot	One screenshot	Picture
Flow chart	Shows how data is processed within the IMS, which parts are involved.	Picture
Evaluators	Name(s) and organisation(s) of the evaluator(s)	Text
Date of contribution	Date(s) of the evaluation of the IMS and of the contribution	Date

Table 1: Suggested structure of the database on IMS

4.3 Structuring Identity Management Systems from the Product View

As an example some IMS identified in [ICP03] are categorised according to the proposed structure. The list of systems from the study was updated; a number of products had to be deleted and several ones were added. This list is by no means comprehensive, but gives an impression of the diversity of IMS.

4.3.1 Class 1 IMS

Subclass	***Product***	***Comment***
Standards	Liberty Alliance[5]	See Chapter 5.1 in this document
	SOAP[63]	See Chapter 5.2 in this document
	OASIS XDI/XRI[6]	XML-based description for identity information
	OpenPrivacy[7]	Distributed user profiles
	vCard[8]	IETF-specified MIME type for business cards
	HR-XML[9]	XML schemata for transfer of human resources information
Applications: Cookie Management	CookieCooker[10]	HTTP cookie manager, interacts with JAP
	CookiePal[11]	HTTP cookie manager
	Privoxy[12]	HTTP proxy with cookie management features

[5] http://www.projectliberty.org/
[6] http://www.xdi.org/
[7] http://www.openprivacy.org/
[8] http://www.ietf.org/rfc/rfc2426.txt
[9] http://www.hr-xml.org/
[10] http://www.cookiecooker.de/
[11] http://www.kburra.com/cpal.html
[12] http://www.privoxy.org/

Subclass	***Product***	***Comment***
	Mozilla Cookie Manager	Integrated browser feature for cookie management
Applications: Social Networking	Orkut[13]	Web-based social networking
	FOAF[14]	XML/RDF crossreferences
	Friendster[15]	Web-based social networking
	OpenBC[16]	Web-based social networking for business contacts
	LOAF[17]	Additional e-mail header to find intersections of e-mail addresses books
Applications: Partial Identity Management	iManager[18]	See Chapter 5.4 in this document
	Light-weight Digital Identity[19]	Personal data are managed by a CGI script which checks authorisation
	Sxip[20]	Partially distributed identity management system
	DRIM[21]	Privacy enhancing identity management
Applications: Password Management	Roboform[22]	Password manager and form filler
	Norton Password Manager[23]	Password manager and form filler
Technologies: Single Sign-On	CA eTrust[24]	Corporate access management

[13] http://www.orkut.com/
[14] http://www.foaf-project.org/
[15] http://www.friendster.com/
[16] http://www.openbc.com/
[17] http://loaf.cantbedone.org/intro.htm
[18] http://www.iig.uni-freiburg.de/telematik/atus/idm.html
[19] http://lid.netmesh.org/
[20] http://www.sxip.com/
[21] http://drim.inf.tu-dresden.de
[22] http://www.roboform.com/
[23] http://www.symantec.com/passwordmanager/
[24] http://www3.ca.com/Solutions/Overview.asp?ID=4839

Subclass	***Product***	***Comment***
	MS Passport[25]	Global Centralised SSO
	Kerberos[26]	Centralised SSO
	AssureAccess[27]	SSO for HTTP and Java 2 Platform
	ClearTrust Federated Identity Manager[28]	SOAP/SAML based SSO
	RSA Nexus[29]	Status (availability, prototype etc.) unknown
	SecureAccess[30]	Enterprise SSO solution
	PingID[31]	SAML-based
	Shibboleth[32]	Web single sign-on using XMLSig and SAML
	Oracle COREid[33]	Part of the Oracle Identity Manager for SSO and web access control
	CIDAS[105]	Central authentication and authorisation solution offering SSO, management of different levels of authentication and anonymous authentication
Technologies: Infrastructure, Third Party Services	X.509 CAs	hundreds of independent roots of namespaces
	X.509 Standard	Standard for public key certificates in X.500 directories
	KeyNote Standard[34]	Allows delegation of authorisation to other keys
	SDSI/SPKI Standard[35]	Defines local namespaces which can be chained by public key crypto

[25] http://www.passport.net/Consumer/default.asp
[26] http://web.mit.edu/kerberos/www/
[27] http://www.entegrity.com/products/aa/aa.shtml
[28] http://www.rsasecurity.com/node.asp?id=1193
[29] http://www.rsasecurity.com/go/NEXUS/
[30] http://www.proginet.com/products/securaccess/securaccess_overview.cfm
[31] http://www.pingidentity.com/
[32] http://shibboleth.internet2.edu/
[33] http://www.oracle.com/oblix/integration.html
[34] http://www.faqs.org/rfcs/rfc2704.html
[35] ftp://ftp.isi.edu/in-notes/rfc2693.txt

Subclass	***Product***	***Comment***
	spamex[36]	Forwards e-mail, hides receiver
	mixmaster[37]	Anonymising remailer
	mixminion[38]	Anonymising remailer with advanced features
	spamgourmet[39]	Forwards e-mail for a user-supplied number of e-mails, hides receiver
	the identity network[40]	Provides services for federation of identities
Technologies: Physical Delivery Intermediaries	None	
Technologies: Payment Intermediaries	eGold[41]	Not a bank, payment in gold certificates
	paypal[42]	Money transfers between e-mail addresses

Table 2: Class 1 IMS

[36] http://www.spamex.com/

[37] http://mixmaster.sourceforge.net/

[38] http://mixminion.net/

[39] http://www.spamgourmet.com/

[40] http://www.pingid.net/

[41] http://www.egold.com/

[42] http://www.paypal.com/

4.3.2 Class 2 IMS

Subclass	***Product***	***Comment***
Communication Management / Workgrouping	PGP[43]/GnuPG[44]	E-mail encryption software
	Ciphire[45]	E-mail encryption software with e-mail addresses as only ids
	OpenSSL[46]	Encryption tool and library, implements S/MIME
	Hushmail[47]	Webmail provider with encryption
	Cryptomail[48]	Webmail provider with encryption, Open Source
Shop Systems / Auction Systems / Reputation Systems	ebay[49]	Online auction, lists reputation of buyer and seller
	advogato[50]	Community website, articles are scored by the reputation of the author.
	Aura[51]	Open Source library for reputation handling
	Slashcode[52]	Community reviewed webpages, e.g., slashdot.org, uses reputation
Online Games	The Sims online[53]	Online game
	There[54]	Meeting place for avatars

Table 3: Class 2 IMS

[43] http://www.pgp.com/
[44] http://www.gnupg.org/
[45] https://www.ciphirebeta.com/
[46] http://www.openssl.org/
[47] http://hushmail.com/
[48] http://www.cryptomail.org/
[49] http://www.ebay.com/
[50] http://www.advogato.org/
[51] http://www.geekness.net/tools/aura/
[52] http://www.slashcode.com/
[53] http://thesims.ea.com/index_flash.php
[54] http://www.thereuniverse.com/

4.3.3 Class 3 IMS

Subclass	***Product***	***Comment***
Browsers:	Mozilla[55]	Password/cookie manager and form filler included
	Opera[56]	Dto.
	Internet Explorer[57]	Dto.
Chat clients	Jabber[58]	Supports multiple nicks and optional encryption
	gaim[59]	Supports multiple nicks

Table 4: Class 3 IMS

[55] http://www.mozilla.org/
[56] http://www.opera.com/
[57] http://www.microsoft.com/windows/ie/
[58] http://www.jabber.org/
[59] http://gaim.sourceforge.net/

5 Designs of IMS

This chapter describes exemplarily five different designs of IMS. Firstly, with the Liberty Alliance approach and Web Services such as WS-Federation two designs for federated identity management are presented. Then a case study shows the variety of components of a type 1 IMS in practice, elaborating also on tasks of the different administrator roles. Finally two research prototypes for type 3 IMS functionality are depicted which aim at supporting the user's privacy: While the *iManager* focuses on usability of identity management, also in a mobile context, and is implemented on a PDA, the *idemix* credential system enables unlinkability for the user's credentials by cryptographic means.

The presented examples were chosen to show a broad variety of designs on the market and in research labs, but they cannot give a comprehensive overview of all IMS designs. Further examples can be found in [ICP03].

5.1 Liberty Alliance and Sun Java Access Manager

This section gives an overview of the Liberty Alliance and details one conforming implementation named Access Manager. The Access Manager is one component of the Sun Java Enterprise System.

5.1.1 Liberty Alliance

The goal of the Liberty Alliance Project[60] is the development of an open standard for federated network identity. The alliance includes companies, non-profit and government organisations as members and the total number exceeds 150.

Different kinds of memberships are possible, according to the level of involvement and the available budget (management board members, sponsor members, associates and affiliates).

5.1.1.1 Goals

The following five expert groups are developing the specifications [Sun05]:

- Technology (development of sample implementations and interoperability tests)
- Public Policy (regulatory issues, legal compliance, …)
- Business & Marketing (identification of market requirements)
- Conformance (interoperability and conformance testing)

[60] Liberty Alliance Project Website, http://www.projectliberty.org/, February 2005.

- Services (identity service specifications)

5.1.1.2 Status

To date, several case studies have been conducted, whitepapers and guidelines have been made available and a number of specifications are available for download on the project's website. Based on these specifications and guidelines many vendors have implemented solutions which are now available in the market.

Recently, Sun Microsystems has successfully conducted an authentication trial with 80 million users. This trial was based on the Java System Access Manager which is described in detail in the following section.

5.1.2 Sun Java System Access Manager

The Sun Java System Access Manager [Sun05] (previous versions of which were known as "Sun Java System Identity Server" and "Sun ONE Identity Server") is a part of Sun's Identity Management framework. It provides functionality to manage access to resources by providing mechanisms for single sign-on (SSO) as well as the main building blocks of an identity management system:

- Authentication, Authorisation and Accounting/Auditing across multiple servers;
- a centralised administration with capabilities for delegation;
- the concept of *Federated Identity* supporting standards such as the Security Assertion Markup Language (SAML) and the Liberty Alliance specifications;
- a highly scalable identity directory.

The foundation for an identity platform is laid by five main components, namely the Sun Java System Directory Server and four components integrated into the Access Manager, as shown in Figure 1:

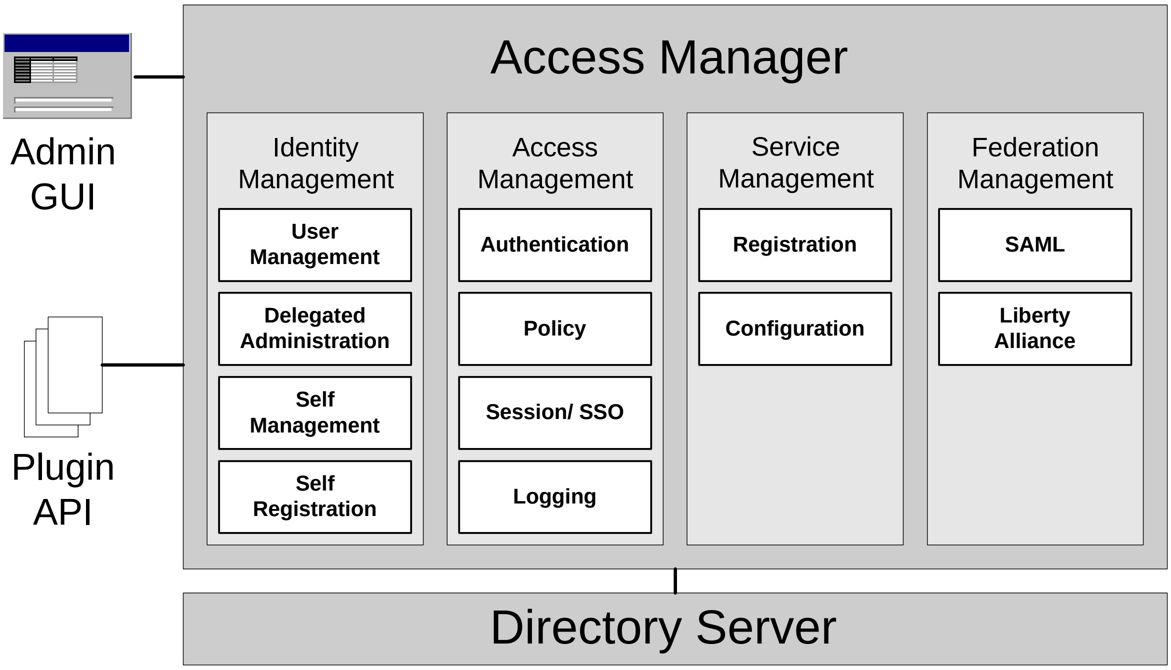


Figure 1: Components of the identity platform

- The ***Sun Java System Directory Server*** is an LDAP-based central repository for identity, application, and network resource information. In the context of identity management, it is used for storing and managing information related to identity profiles, access privileges, and policies.
- The ***Identity Management*** component supports administration tasks such as managing users' identities, services and polices, by providing tools and GUIs that may be used to customise and automate the related tasks. To reduce the administrative overhead in systems containing a large number of participating users, the users may be enabled to manage their own data, or parts thereof, via this component (*Self Management/ Self Registration*). Delegated Administration is supported by Role-Based Access Control mechanisms.
- The ***Access Management*** component provides infrastructure for authentication and authorisation tasks, allowing the centralised enforcement of access control policies for multiple resources using a single account for each user. SSO mechanisms allow the user to access resources on multiple servers without having to authenticate repeatedly for each new resource. Cross-Domain SSO (CDSSO, see Section 5.1.2.1) allows SSO across multiple different DNS[61] domains. The Access Management component supports multi-level authentication, where each authentication level corresponds to at least one authentication mechanism. Authentication levels are assigned to all resources, and the user may choose from the associated authentication mechanisms when authenticating for a resource. After a successful authentication, the user will only have to re-authenticate for resources a higher authentication level has been assigned to. Policy agents integrate application servers and web servers with the

[61] DNS: Domain Name Service; service for name-IP address resolution use in the Internet

Access Manager: Whenever a user attempts to access a protected resource via the web server, the respective policy agent determines whether an authentication token is present, and redirects the request to the Access Manager for authentication if necessary.

- The ***Service Management*** components provides GUIs and tools for the administration of services, including tasks such as registering services or updating service attributes. In this context, a service is abstractly defined by a name and a group of attributes describing related information, i.e. typically but not necessarily the parameters provided by a service actually implemented as a software module. The services of the Access Manager themselves (the Core Services providing its basic functionality) are configured via this component as well.
- The ***Federation Management*** component provides functionality for SAML interoperability and federated identity. Federation is a term describing the linking of a user's separate accounts across multiple domains. To achieve federation, information about the respective user must be exchanged securely. The SAML standard is used in this context for exchanging security assertions between trusted security authorities.

5.1.2.1 Cross-Domain Single Sign-On

An essential functionality of the Access Manager component provides mechanisms for SSO. With SSO, a user that has been authenticated to the Access Manager may use any application managed by the Access Manager without having to re-authenticate. In the following, we consider web-based SSO only. In this scenario, the user interacts with the Access Manager authentication interface via his browser, which holds a cookie containing authentication credentials, i.e. the user's session identifier, after a successful authentication. Whenever the user tries to access a resource protected by the Access Manager, the information within the cookie is used by the protected resource to determine whether the user may be granted access to the resource. For security reasons, however, cookies are assigned to specific domains and may subsequently only be accessed by servers in the respective domain. Therefore problems arise when the Access Manager is used to manage applications within different domains, because the information required to grant access to a user is not available to applications within different domains.

In order to enable Cross-Domain Single Sign-On (CDSSO), the Access Manager provides a CDSSO component issuing cookies for a specific domain. Therefore, all different participating domains require a CDSSO component, which is in each case installed as a servlet into a web server of the respective domain. A complete CDSSO authentication process contains a rather large number of necessary steps, most of which are hidden from the user. Once the authentication is complete, however, subsequent request are processed in a straightforward manner without the necessity of cross-domain redirects.

The following components participate in a CDSSO authentication process, as shown in Figure 2.

- The ***browser*** representing the client requesting authentication;

- The ***web server*** representing the protected resource the browser attempts to access, including an ***SSO agent*** protecting the resources accessible via the web server;
- The ***CDSSO component*** (located in the same domain as the web server) issuing authentication cookies;
- The ***Access Manager*** (located in a different domain than the web server) including a ***CDSSO servlet*** acting as a broker between components.

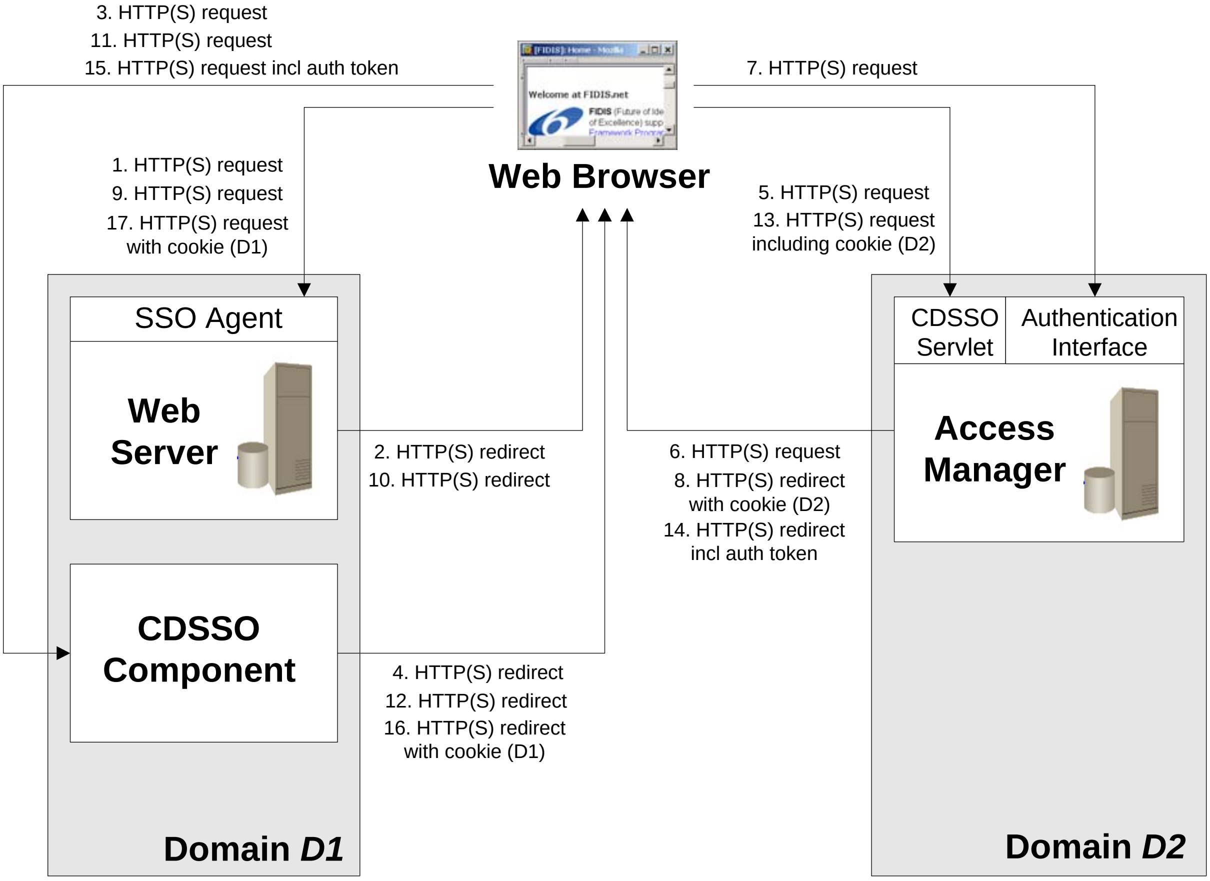


Figure 2: A Cross-Domain SSO authentication process

A Cross-Domain SSO authentication process

The CDSSO authentication process contains the following five steps, as shown in Figure 2:

- The initial request by the browser to access a resource in the domain *D1*. This request is sent to the web server of the respective domain *(step 1)*. The SSO agent checks the request for an authentication token and redirects the request to the CDSSO component *(steps 2, 3)*, because an authentication token is not present.
- The redirect to the authentication interface in the domain *D2*. The CDSSO component redirects the request to the CDSSO servlet of the Access Manager *(steps 4, 5)*. Again, the request is checked for an authentication token and redirected to the Access

Manager authentication interface *(steps 6, 7)*, because the token is not present. The user now authenticates successfully.

- The redirect back to the web server representing the protected resource. After successful authentication, a cookie is stored for the domain *D2* and the browser is redirected to the resource that originally had been attempted to access *(steps 8, 9)*. The SSO agent checks the request for an authentication token one more time and redirects the request to the CDSSO component again *(steps 10, 11)*, because there is only a token present for the domain *D2*, but not for the domain *D1* of the protected resource.
- Another redirect to the authentication interface in the domain *D2*, similar to the second stage. The CDSSO component redirects the request to the CDSSO servlet of the Access Manager *(steps 12, 13)*. This time, an authentication token for the domain *D2* is present. The request is redirected to the CDSSO component in the domain *D1 (steps 14, 15)*. The authentication token is included as a URL parameter in the request and used by the CDSSO component to set a cookie in the domain *D1*. Then, the request is redirected for the last time to the web server representing the protected resource *(step 16)*.
- The process completion. The browser finally requests to access the requested resource *(step 17)*. The cookie for domain D1 includes the authentication token, which the SSO agent validates successfully via the Access Manager *(not shown in* Figure 2*)*. Access to the requested resource is granted.

5.1.2.2 Chaining of Authentication Mechanisms

Another aspect supported by the Access Manager is the possibility to configure authentication modules in a way that the user must provide authentication credentials to different modules within a chain. The modules within the chain are used, one by one, to process an authentication step that may either succeed or fail. The entire authentication process is terminated successfully when the end of the chain is reached, the last module in the chain succeeds to authenticate and no modules marked as “required” have failed to authenticate. Furthermore, the process is terminated successfully immediately when a module marked as “sufficient” succeeds in authenticating. Additionally, the process is terminated unsuccessfully immediately when a module marked as “requisite” fails to authenticate. Modules marked as “optional” do not have an additional effect on the overall authentication process. Consider the following scenario: three website are available via SSO: an information site, a shop site and a banking site. All websites use the same two authentication modules, M1 (using a username/ password mechanism) and M2 (using a biometric authentication mechanism) within a chain.

M1 is marked as “required” for all three sites, while M2 is marked as “required” only for the banking site, and as “optional” for the other site. A user may therefore log into the information site via M1. An authentication token is generated when the authentication succeeds, and the user may access the information. When accessing the shop site, the user does not have to re-authenticate because the authentication token is used for SSO. When accessing the banking site, the authentication token indicates that the user has authenticated via M1. However, because the biometric authentication via M2 is required for the banking site, the user has to re-authenticate.

5.1.2.3 Supported Standards

The Sun Java System Access Manager supports the following standards:

- JAAS (Java Authentication and Authorisation Service)
- Liberty Alliance Phase 2 (Identity-based Web Services Framework)
- OCSP (Online Certificate Status Protocol)
- SAML 1.1 Specification
- SOAP (Simple Object Access Protocol) 1.1
- SPML (Service Provisioning Markup Language)
- SSL (Secure Sockets Layer)
- XML Digital Signature
- XML Encryption

5.1.3 Summary

After an introduction to the open standardisation group Liberty Alliance, this section has shown components, architecture and functionality of the Sun Java Access Manager as type 1 IMS, comprising authentication, authorisation and accounting. A trial with 80 million users were used to test the presented configuration of the Sun Java Access Manager.

5.2 Web Services, the Claims-Based Security Model and Federated Identity Management

5.2.1 Introduction

In today's world, more and more systems and applications communicate through so called 'web services[62]'. Generally speaking, the term 'web service' refers to XML-based messaging via internet protocols, such as HTTP or SMTP. The *'Simple Object Access Protocol'* (SOAP[63]) is the XML based protocol that provides a definition how structured and typed information can be exchanged between peers in a distributed and decentralised environment.

In order to provide security, reliability, transaction abilities and rich metadata support, additional specifications exist on top of the XML/SOAP stack. The core technologies in the XML space are all defined by the *World Wide Web Consortium* (W3C). Many of the higher-layer specifications are driven by multiple industry players, including Microsoft, IBM, BEA Systems, SAP and others. These specifications are helping to progress interoperability between the different industry platforms.

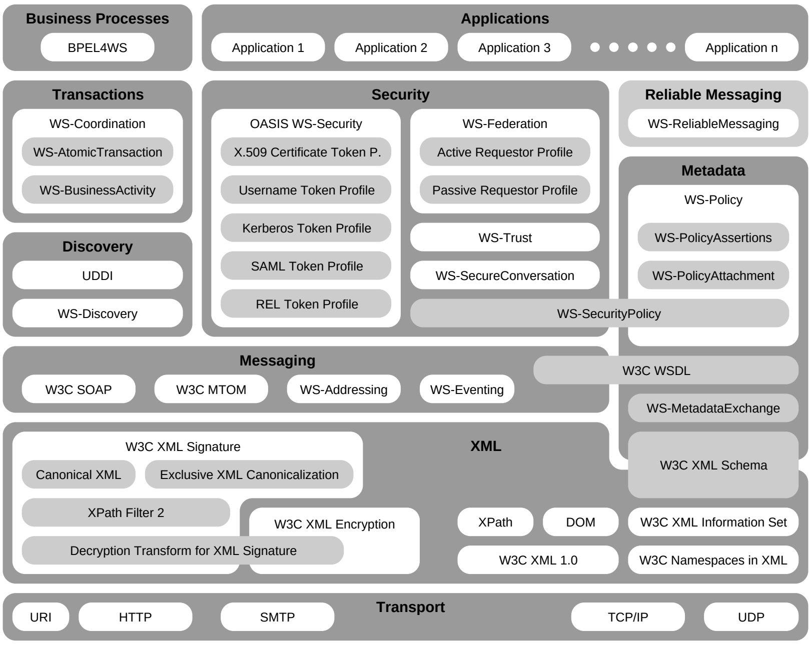


Figure 3: Overview of different web services specifications

The main theme across all these different specifications is that they are orthogonal to each other, so that they contribute to a composable architecture. Instead of having a single

[62] Luis Felipe Cabrera, Christopher Kurt, and Don Box. An Introduction to the Web Services Architecture and its Specifications, version 2.0. October 2004. http://msdn.microsoft.com/library/en-us/dnwebsrv/html/introwsa.asp

[63] W3C Simple Object Access Protocol (SOAP), Version 1.2. June 2003. http://www.w3.org/TR/soap12

framework that prescribes everything from network protocols and message exchange patterns down to data formats, the specifications in the industry roadmap provide a rich set of tools to provide security, reliability and transactability for the web services environment.

5.2.2 WS-Security

The SOAP communications infrastructure requires additional security mechanisms for implementing the various security services. Simply speaking, the WS-Security specification[64] provides mechanisms to attach so-called ‘security tokens’ to a message and to sign and/or encrypt the message or parts thereof. Thus, WS-Security provides message-based security that is independent of the security properties of the underlying transport channel. A security token represents a collection of one or more so-called ‘claims’. The receiver of a SOAP message or intermediary nodes can use these security tokens to validate that certain assumptions are correct. A ‘claim’ is a statement about an entity (like a principal, client, a service or a resource), whereas this statement itself could be a name/ID, cryptographic key, group membership, privilege, capability or an authorisation statement.

Example: *SOAP sender could digitally sign the payload (the SOAP Body element) of the message with his private key and attach the associated X.509 certificate as security token in a SOAP Header. A SOAP intermediary like a web service firewall or the final receiver of the SOAP message could then extract the security token from the received message, validate that the claims inside the security token are valid, and validate that the security token was used to create the digital signature of the message.*

WS-Security in itself defines how arbitrary security tokens can be attached to a SOAP message and how confidentiality, integrity protection and authentication mechanisms can be applied to the message, e.g. where to put the digital signature. WS-Security does not constrain what particular security tokens can be used, that is left to the various security token profile specifications. These profiles cover a wide range of different security token formats, ranging from X.509 certificates, username/password tokens, Kerberos tickets or SAML assertions to REL tokens (ISO Rights Expression Language). In the case of X.509 or Kerberos, existing binary tokens are wrapped inside XML. Besides these standardised token formats, the system is extensible so that arbitrary custom XML-based tokens can be used together with WS-Security.

WS-Security does not prescribe what particular token types have to be attached to messages, nor does it define from where these security tokens originate from. A security token could be taken from local token repositories (such as a local certificate store), or they can be retrieved from specialised security token services, such as an X.509 CA[65], a Kerberos KDC[66] or a WS-Trust security token service.

[64] OASIS Web Services Security SOAP Message Security 1.0 Specification. March 2004. http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf
[65] CA: Certificate Authority
[66] KDC: Key Distribution Centre

5.2.3 WS-Trust

The claims inside a security token can represent arbitrary assertions, in which some entity makes some statement about some other entity. For example, in an X.509 certificate, a CA states that a particular public key belongs to a particular entity, whereas that entity is identified by a given name. Depending on the application's security and privacy requirements, a security token may include very specific information. In the web services world, web service endpoints can express message security requirements (using WS-SecurityPolicy).

Example: *A web service may require that an incoming message must prove a set of claims, e.g. the message must be signed with a security token issued by a specified token issuer. Messages that do not comply with this requirement will be ignored by the service. In order to comply with that requirement, the message sender may have to retrieve a sufficient security token from a token issuer.*

The WS-Trust specification[67] defines a web services based interface for requesting, issuing, renewing and validating arbitrary security tokens. WS-Trust calls such issuers 'security token services' (STS). In most real world cases, a client will send a SecurityTokenRequest message to an STS, along with certain security claims and a proof of possession, and then expect that the STS sends back another security token that contains the desired claims. Such exchanges can be done multiple times with different security token services. Each of these services receives a set of claims, validates these claims and issues a new security token for the client. Simply speaking, a WS-Trust STS can be seen as a token translator, translating one type of information into another one. An STS may also act as a proxy for token requests or validation operations. For instance, an STS may query another STS if it cannot perform a validation on itself.

5.2.4 WS-Federation

The WS-Federation specification[68] introduces mechanisms to manage and broker trust relationships in heterogeneous and federated[69] environments. This includes support for federated identities, attributes and pseudonyms. 'Federation' refers to the concept that two or more security domains agree to interact with each other, specifically letting users of the other security domain access services in the own security domain. For instance, two companies that have a collaboration agreement may decide that employees from the other company may invoke specific web services. These scenarios with access across security boundaries are called 'federated environments' or 'federations'. Each security domain has its own security token service(s), and each service inside these domains may have individual security policies.

[67] WS-Trust Specification. May 2004. http://msdn.microsoft.com/ws/2004/04/ws-trust/

[68] WS-Federation Specification. July 8 2003. http://msdn.microsoft.com/ws/2003/07/ws-federation/

[69] IBM Corporation and Microsoft Corporation. Federation of Identities in a Web Services World Whitepaper. Version 1.0. July 8, 2003. http://msdn.microsoft.com/ws-federation/

WS-Federation uses the WS-Security, WS-SecurityPolicy and WS-Trust specifications to describe patterns and scenarios that allow entities from the one domain to obtain security tokens valid in the other domain, thus subsequently getting access to the services in the other domain.

Example: Michael, *an employee of Contoso Ltd. wants to purchase goods at a Fabrikam's shop web service. The employee privacy policy of Contoso Ltd. does not permit that PII (personal identifiable information) from employees is exposed to an outside partner such as the other company. It also would be an administrative nightmare to make all identities of authorised employees from Contoso available to the shop and to keep that list up to date. Additionally, the employee's identity inside Contoso will probably be meaningless to the Fabrikam shop, because the shop is in another trust domain.*

The solution is to have WS-Trust exchanging existing security tokens into different ones that are understood by the other party:

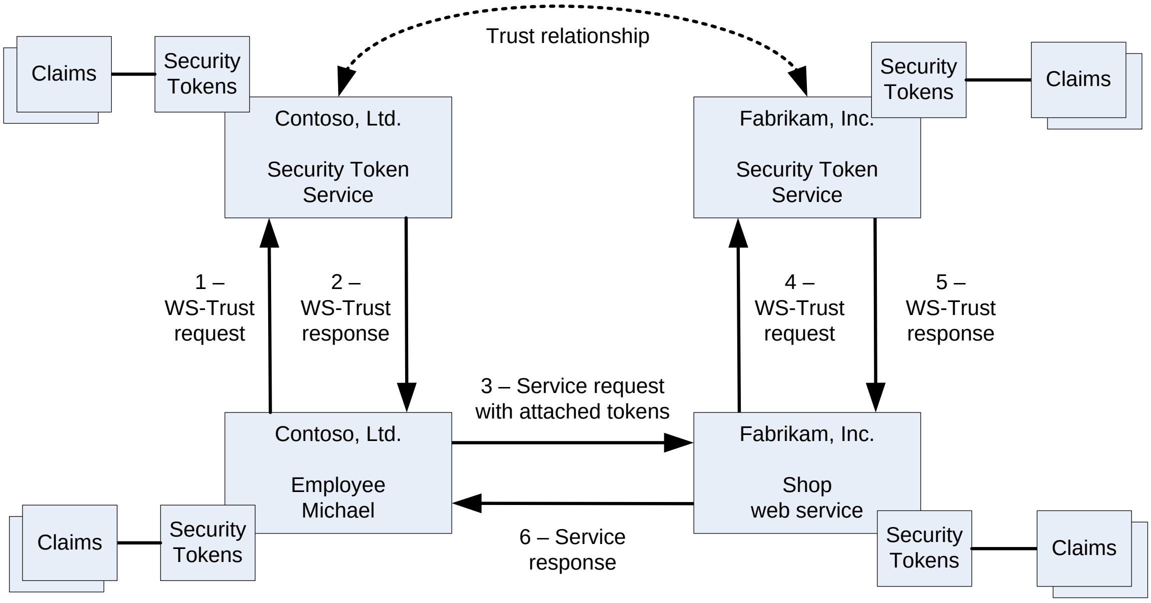


Figure 4: WS-Trust/WS-Federation example

The employee's shopping application sends a SecurityTokenRequest message to Consoto's STS, requesting a token that will be accepted at Fabrikam (step 1): "I am employee Michael from our IT department and I have to invoke the Fabrikam shop web service at this and that address." *The claim* "I am employee Michael" *is proved, for instance by signing that request message with the employee's private key or by attaching a valid Kerberos token from Contoso to the message. After internal checks (such as* "Is our employee Michael authorised to buy at Fabrikam's shop of behalf of Contoso?"*), the STS from Contoso sends back a security token to Michael (step 2). That security token may include two things: It must include (a) a security token that Michael can attach to the web services invocation and it may include (b) a proof token for Michael with which he can prove that he possesses the security token. The security token may state that* "The entity possessing this or that cryptographic key is an employee of

Contoso and may purchase goods on our behalf". *This security token is attached to the shop message (step 3). 'Attached' means that the shopping request is signed with the token. The shop's web service cannot interpret the semantics of the token, and asks the Fabrikam's STS to validate the token (step 4). The STS from Fabrikam may return a new security token containing information like:* "The owner of this or that cryptographic key is employee of one of our gold customers" *(step 5). After checking that the message from step 3 is signed with the specified cryptographic key and that the security token from step 5 contains a claim that access is granted, the shop request is executed and the results are sent back to the customer (step 6).*

5.2.5 Summary

The security-related web services specifications WS-Security, WS-Trust and WS-Federation introduce a model for passing arbitrary types of security claims and tokens between communicating peers. Many of the existing systems scale very well inside closed corporate environments but, like Kerberos, do not work cross-organisationally and require high management overheads for user management (such as username/password) combinations. Claims-based security enables systems to utilise security tokens that may link the owner of a certain cryptographic key to a certain group, without revealing the identity of the owner itself.

The claims-based security model enhances the conventional methodology by abstracting away the token type and by only defining patterns and semantics for the exchange of arbitrary token types, thus enabling cross-trust boundary scenarios as well as distributed peer-to-peer models that are not possible or are unmanageable with today's systems.

The claims-based security model eases information security management by distributing responsibility for managing identity information to the relevant parties. In the previous example, we have seen that the company is responsible for managing its own user database (to validate company-internal tokens) and for authenticating other organisations, whereas authenticating or authorising a principal from another organisation is done by that other organisation itself.

5.3 Case Study: Enterprise Identity Management in a Bank

This is an extracted content summary of the case study that describes the bank project implementing the introduction of the Enterprise Identity Management (EIM) system in its entirety. The case study was conducted by Budapest University of Technology and Economics.

5.3.1 Premises

The monetary institution that holds an important position in the Hungarian bank market started its BCP-DRP (Business Continuity Plan – Disaster Recovery Plan) program in 2001, and updating the identity management systems of the bank was part of this program. The justification of the project and the biggest problems of the system operating back then can be summarised in the following:

1. Positions and assignments weren't properly matched, and bank authorisations assigned to positions weren't set up consequentially. (For example: the same positions could have different authorisations in specific bank offices.) The lack of regulation was apparent in the case of both the existing bank applications and the administrative procedures in force.
2. The various bank systems were set up with different authorisation management solutions whose maintenance and updating can be realised only with difficulties.
3. Possible queries concerning the past activities of users/employees didn't comply with the requirements of BankSecurity in all respects.
4. The BankSecurity had only incomplete information about the current state of authorisation management.
5. The lengthy authorisation management procedures (for example, application for, or the modification of authorisation) decrease the efficiency of both the IT and the business fields.

5.3.1.1 The Structure of the Project

The project aimed at the introduction of the EIM system can be divided into four phases:

1. Feasibility – 2002
2. Pilot project – 2003
3. Implementation phase I – 2004
4. Implementation phase II – 2005

5.3.2 Feasibility

The bank decided on commissioning a feasibility study since the executives were concerned about the heterogeneous informatics environment and the domestic singularity of the project. The following possible solutions crystallised as the result of the work completed by a firm of external advisors:

1. *Reconsidering, reshaping and regulating the existing identity management*
   Besides leaving the existing – fragmented – identity management solutions in place, the security level could be raised through the changes in identity management. According to this version, setting up a role-based user management, rethinking the identity management procedures and introducing a uniform regulation procedure are all necessary.
2. *Development of an in-house application to support critical fields*
   Several current problems of authorisation management could be resolved by applications developed in-house. These low cost developments include: supporting the administrative procedures of authorisation management and producing certain statistics about accesses. Subsequent changes resulting from new regulations in the most important bank systems also helped the proposed process.
3. *Introduction of the EIM application*
   Introducing enterprise identity management requires completing the tasks detailed under point 1. An enterprise identity management level is one in which the tasks of change and administration management are resolved in a centralised way, and the supervision of execution is also done at one place, with the help of an informatics tool. The introduction of a enterprise identity management system – following a proper preparation – provides several options for both the employees of the bank and the BankSecurity field that takes over the tasks of identity management, which, in turn, besides improving security may result in the simplification of the administrative processes. The most important features of the enterprise identity management system support applications are: Single Sign-On, authorisation automation, rule-based authorisation management, PKI support, a database that can be queried by SQL, user database, authorisation management workflow, integration of bank security systems, card-based identification, intrusion detection, response-reaction activation.

The table below displays what priority the introduction of the individual services has for the bank, furthermore whether the different realisation methods to be described provide satisfactory solutions for the given requirement.

Description of requirement	Priority	Solution *		
		1	2	3
Single Sign-On (SSO)	High			✓
Modification of authorisation procedures	High	✓	✓	✓
Newly logged in user to access everything immediately	Medium			✓
Rule-based authorisation management	High		✓	✓
PKI	Medium			✓

		Solution *		
Scalability	Medium		✓	✓
Database and Log files (dynamic database) that can be queried by SQL	High			✓
Support of paper-based procedures by electronic applications	High		✓	✓
Every access within the new system	High			✓
Integrating the bank security systems	Medium			✓
Realisation of administration with 4-6 people	Medium		✓	✓
Support of exception management	High		✓	✓
Regulation of administrative intervention	High	✓	✓	✓
Card-based identification	Medium			✓
Intrusion detection	High			✓
Response-reaction activation	High			✓
Dynamic authorisation portfolio	Medium		✓	✓
External sign-in	Low			✓
Option to modify own password	Medium		✓	✓
Web interface	Low			✓
Option for mass data modification	High		✓	✓

*Explanation: 1st column: The reconsideration, reshaping and regulation of the already existing authorisation management; 2nd column: Development of an in-house application to support critical fields, 3rd column: Introduction of a enterprise identity management application.

Table 5: Description of requirements

Following the analysis of the solutions provided by the realisation alternative solutions and the expectations of the bank, it has been concluded that the bank can fulfil all the expectations through the introduction of the enterprise identity management system. The introduction of an application supporting the enterprise identity management system and the services it provides will, however, provide real advantages for the bank if multi-user systems (an account-management system, for example) also get integrated.

5.3.2.1 Participants of the EIM Software Market

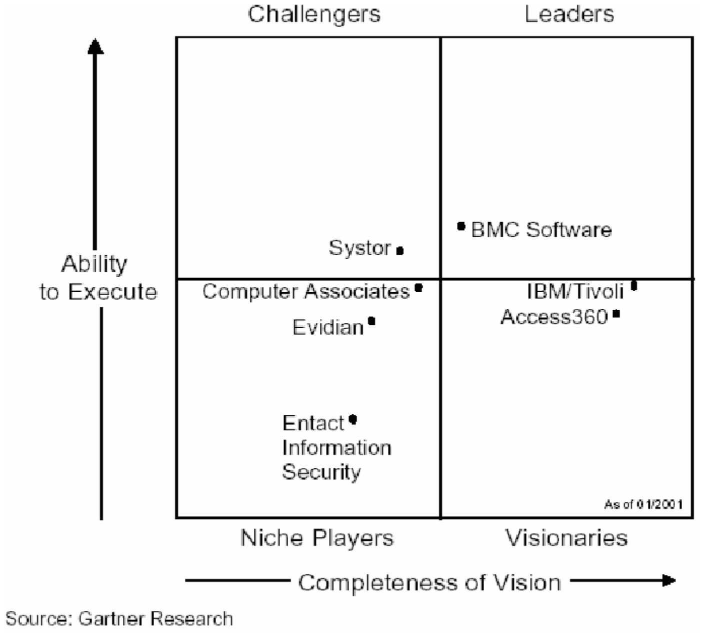


Figure 5: Overview of participants of the EIM software market

AccessMaster

AccessMaster is a product of the Evidian (previously BullSoft) that combines administrative functions with an SSO application. The integration of user administrative and SSO applications made the coordination and the combined management of the data of the generic user database and the user identifier possible. AccessMaster is capable of administering both web-based and non-web based applications just as well applications of remote access. In 1999 Evidien, as the only European provider, initiated investments in South America as well[70].

Control-SA (CTSA)

Control-SA originally is a development of EagleEye, a subsidiary of New Dimension that is now a part of BMC Software. It is highly reliable software that provides a wide range of services besides the administrative ones. A further advantage of it's that it supports numerous operating systems, mainframes and applications. BMC previously made a deal with two EAM (external access management) application manufacturers to gain their support for the administration of web-based applications[71].

Entact!

Its developer is Entact, a new entrant of the EIM market. It's known mostly in Europe thanks to both its mainframe-based user administering and reporting (Entact!Admin) and audit ((Entact! Audit) products. A while ago it has entered a partnership agreement with Securant, which product lends support to web-based user administration and SSO[72].

[70] See http://www.evidian.com/accessmaster/sso/
[71] See http://www.bmc.com/products/proddocview/0,2832,19052_19429_22855_1587,00.html
[72] See http://www.entact.com/

ETrust Admin

ETrust Admin is a product of Computer Associates, a result of the integration of the address-database service and the security management technology of the company. It supports the administration of both web- and non-web based users[73].

Microsoft Metadirectory Services

In July, 1999 Microsoft bought ZOOMIT Corporation that was a market leader of Metadirectory solutions back then. As a result of the acquisition Microsoft provides a comprehensive platform for the Microsoft Windows 2000 Server that includes the security, Active Directory and the metadirectory services as well[74].

Security Administration Manager (SAM)

The EIM product of Systor (formerly Schumann-AG). The included Role Miner supports rule-based access effectively. It can be comfortably used even in a heterogeneous software and hardware environment, thanks to the development of the Agents that became possible upon the SAM Connect interface's becoming openly accessible[75].

Tivoli User Administration and Security Manager

This is a product of IBM that combines user administration and security management applications. Its biggest fault is its weak support of heterogeneous platforms. Its implementation requires the Tivoli Framework. This situation will probably change in the future as development pointing towards the complete separation of the two products has already begun[76].

5.3.3 Pilot Project

According to the results of the feasibility study the bank decided on the pilot introduction of the Control-SA solution on the bank client-identification system.

The Control-SA solution of the BMC Software company centralises identity management "under a shield" that makes it possible to administer and supervise all the corporate security systems simply and automatically at a central location. In the planned identity management architecture of the bank this "shield" may be provided by the human-political system, the entry system, and the bank applications managed by the Control SA. This solution provides a comprehensive overview for the realisation of the administration requirements of identity and authorisation. The Control-SA provides a solution for external, internal and third party users just as well as for those business partners who require access to applications, databases, platforms and infrastructures within the bank.

Universal Control-SA architecture

The standard Control-SA system consists of two basic components:

[73] See http://www3.ca.com/Solutions/Solution.asp?ID=271

[74] See http://www.microsoft.com/windows2000/technologies/directory/MMS/default.asp

[75] See http://www.e-consultancy.com/knowledge/whitepapers/75038/systor-security-administration-manager-sam-enterprise-user-administration-eua-product-suite.html

[76] See http://www.tivoli.com/news/features/security/

1. **Enterprise Security Station (ESS):** The ESS consists of a server and a central security administration database (data storage). With its own administrative graphical user interface (GUI), it provides central supervision for the managed systems of the entire company. The data storage of the Enterprise Security Station is a copy of the centrally managed local databases.
2. The **Control-SA/Agent software modules** manage the access management services on the various platforms and applications of the company. Security administrators can view, supervise, check and audit, through the GUI, the access-management of the entire company, including operating systems, databases and directories, e-mail and groupware as well as business applications. The Control-SA/Agents communicate between the Control-SA server and the user databases of the company platforms and applications, providing real time synchronisation. The open architecture of the Control-SA provides the exceptional expandability of the system. The Universal Security Administration API (USA-API)) standard ensures the "integratability" of the security systems developed in-house or provided by new developers. Its scalability and its capability to cooperate mutually with unique systems ensure that its security management remains expandable by the inclusion of further applications, for example intrusion detection, physical security, executive information and human systems. Standardisation makes the unified management of applications running on different platforms through the use of the Control-SA graphic user interface.

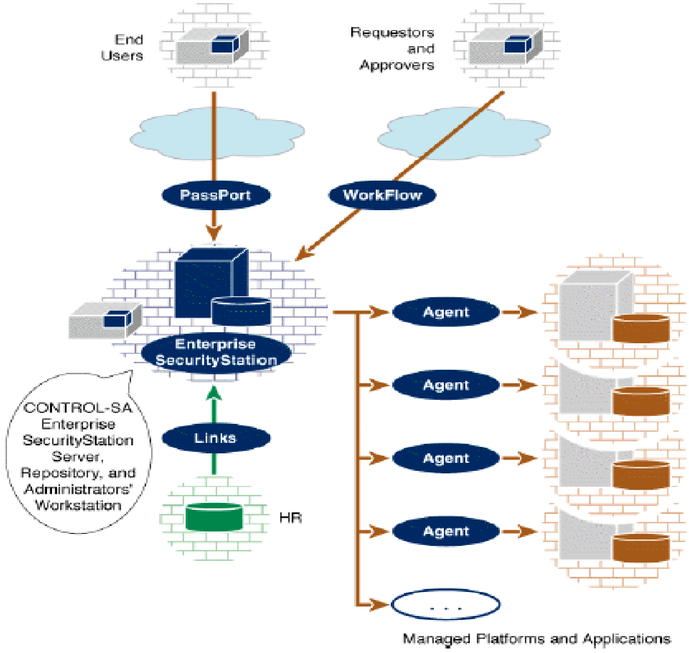

End Users
Requestors and Approvers
PassPort
WorkFlow
Enterprise SecurityStation
CONTROL-SA Enterprise SecurityStation Server, Repository, and Administrators' Workstation
Links
HR
Agent
Agent
Agent
Agent
. . .
Managed Platforms and Applications

Figure 6: Universal control-SA architecture of the planned solution

Synchronising security data

The Control-SA provides a unique, centralised overview of the entire security environment that is both up to date and accurate. The two-way data flow ensures complete synchronisation and compatibility between local security systems and the central data storage of the Control-SA.

Role-based management

The Control-SA has created role-based (profile-based) management for the security administrative activities that can be pursued centrally. The entities of definable position code (enterprise users) represent internal and external user groups that share similar business roles. The position codes and profiles represent positions and assignments within the organisation. Each entity contains those cross-platform authorisations that are required for the fulfilment of certain positions. Assigning a given position code to a given user results in immediate access-rights settings for the user to the resources, which results in increased productivity. By connecting the employee's user ID to the entity, the system provides a complete overview of the authorisation profile. Authorisation can be revoked immediately upon layoff. A given position code can be bound to the access-rights multiple times. Position codes can also be created in such a way that they form hierarchic structures, ensuring greater implementation flexibility. Activities related to the creation or the deletion of a user (creation of user directories, for example) are fully automatic, there is no need for administrative involvement in the process as every variable and value can be imported, calculated or generated according to predefined rules. This advantageous feature excludes human errors and guarantees that the accounts come into being on the managed systems in a consistent way that suits the current company policy. Modification of a profile, in the case, for example, of the introduction of a new application results in the automatic updating of those who belong to this profile. The profile-based management is capable of automatically initiating an immediate and accurate user registration and resource access definition. This simplifies administration, decreases errors and guarantees that all the accounts can be defined on all the platforms in a consistent manner.

Centralised or decentralised management

The Control-SA is capable of providing both centralised and decentralised security administration in view of the given organisational requirements. Security administrators can be appointed as needed according to the organisational structure, the platform, and the operations – or according to any combination of these. These way different persons may define and use different views, be the administrators, operators, auditors or members of the Help-Desk staff. The Control-SA provides a wide range of applications for both the real time observation of operations and report presentation. Each and every operation and entity can be controlled and followed online. Every operation initiated or stopped by Control-SA is verified. Reports about the connections between online entities and verified information can be personalised and planned in advance. These reports can be printed or exported.

Security checks

The corporate level management of Control-SA provides an efficient administration method and a strict, comprehensive control. By the quick and comprehensive overview of the secure environment and all the access rights of the users administrators can follow online and verify who has legal access to what. The Control-SA makes it possible for the security administrator to set up and enforce wide security standards (password structure rules, for example). Violation of these rules on a managed system can be verified and an automatic alert can be generated. Besides this, with the help of the report generator a detailed and personalised audit document is easy to prepare.

Results of the pilot project can be summed up in the following:

i. An *Application map* was developed at the bank that besides providing a realistic snapshot of the general and technical parameters of the various bank applications, also illuminated the possible integration and authorisation issues of the heterogeneous informatics environment.
ii. Regarding the client identification system: authorisation profiles belonging to the various positions were surveyed and rationalised.
iii. During the integration of the client identification system the following was set up:
    i. the interface between the client identification system and the CTSA
    ii. the CTSA/client identification system agent software module that realises the authorisation management function

Following the successful testing phase and the useful documentation, the evaluation of the pilot concluded that the system is eligible for the support of bank identity management functions and tasks, and that with its full implementation the requirements of the bank can be satisfied and the proposed objectives achieved.

Implementation phase I – integrated account management system and the integration of the MS AD/Exchange environments

Objectives:

As a result of the Project, the two most important applications of the bank came under the unified identity management systems, and a direct daily connection is developed towards the HR system and the bank security entry system. Hence the input data of the administrative system also enter the process in an automated way. The employees and contractors entering or shifting positions within the bank automatically gain access to the authorisation profile determined by HR on the managed systems. This increases the security, confidentiality, and availability and audit ability. In the case of the non-managed systems (whose number decreases constantly) the previous user administration remains in operation. The effort in user administration can be decreased, the database relevant for user administration gets set up automatically in a enterprise system. The user administration managed by the EIM system can be taken over by BankSecurity, canceling the current incompatible practice that is also defective from the view of information security. A properly controlled, state of the art authorisation management environment, well supported by informatics appliances, is born for the entire bank application portfolio.

Work phases and result of the first implementation phase:

- In the planning phase a system plan was completed and the specifications were also laid down in cooperation with the bank.
- The bank had nomograms prepared that provide a good overview of the relations among positions / assignments / bank organisations / applications. With the help of these tables it was possible to pinpoint and define those authorisation inconsistencies that could be corrected in the standard authorisation tables to be created in the next work phase.

- The rationalisation of the entire bank application and positions portfolio took place, that is, the definition of position / authorisation profiles was completed. The results were assembled into a standard authorisation table whose progressive maintenance was made a part of the administrative procedures. The standard table can, of course, be mapped onto the registry of the CTSA which makes the prevention of the previous inconsistencies possible.

The purpose of the standard table

The main element of the central authorisation management is the standard table that incorporates the requestable authorisations of all the bank systems (at the moment still through the Felhadmin[77]) that got integrated into the authorisation management system. The purpose of the creation of the authorisation table was to define authorisation combinations (the so-called combined authorisation profiles) that connect different system rights according to a systematisation that assigns users into unambiguous groups. The first such user distribution was developed for the integrated banking account management system, the client identification system and Active Directory. It was based on the classification and definition of positions and assignments relevant to the given positions, all according to the JobCode (so called Hay code). This division comprised of several hundred authorisation combinations whose maintenance would have bogged down the central authorisation management, especially given that the standard authorisation table incorporated each and every system present in the Felhadmin. To make the handling of the central identity management tasks reasonable it became necessary to decrease the depth of the subdivision of the authorisation table, which resulted in the creation of the authorisation standard table, that is, the authorisation combinations according to the tasks within the organisational units.

Standard table: The method of preparation

The definition of the assignments of given organisational units (their scope of duties, that is) was, in the case of several systems, made according to previously conducted surveys, while in the case of the rest of the systems they were described according to the activities queried from the BCP database. The authorisation combinations were included in the standard table according to the current user authorisation data received from the system administrators. The authorisation combinations of the standard table and the descriptions of positions were reconciled either personally with all the fields of special importance or sent out to every specialty field whose scope of duties was included in the standard table. The assignments and tasks listed in the standard table comprise of the non-unique tasks only, therefore required activities of that kind (along with their authorisation requirements) that a single colleague pursues do not get listed in the standard table. Requiring these remains a unique process processed through the Enterprise Security Station ESS Workflow user rights-requesting interface (see Figure 6). Each and every authorisation requested and granted this way will be registered centrally in the database of the ESS, querieable by the authorisation administrators.

[77] The name of the former user administration tool

Interpreting the standard table

The Xs appearing in the standard table define the kind of access a given assignment for a given organisational unit requires in a given system. Every X signifies a separate system right whose name can be found in the first lines of the column of the given X. The assignments listed by the standard table contain the general tasks performed by the organisational units, therefore activities (along with their authorisation requirements) that a single colleague pursues do not get listed in the standard table. Requiring these remains a unique process.

The institualisation of the standard table

The standard table covers the main areas of the bank's operation by listing the tasks performed by the specialties. Since the current authorisation allocations cannot be mapped fully onto standard table because of the accesses required for the uniquely performed tasks, the introduction of the standard table and the uniformisation of the authorisation systems will be performed gradually. It is the task of the BankSecurity authorisation administration to reconcile the current authorisations with the standard table by the progressive inclusion of the systems managed by the EIM. Within this framework it defines and sets the complex authorisation profiles for each and every colleague, reconciling its decisions with the specialties. It also assesses whether a given colleague really needs all the unique accesses that may not fit the complex authorisation profiles.

- In view of the AD/Exchange we developed the connection using the factorial agent software module of the CTSA. Thanks to the installed entry system connection between the HR and the BankSecurity, colleagues entering automatically gain access (hire-fire) to their email addresses and AD authorisations. (Data on current employees are kept by the HR system while those of outside collaborators are kept by the entry system.)
- Authorisation management terms and principles at odds between the integrated account management system and the CTSA were reconciled. The interface between the two systems was developed, and the scripts providing the queries were written up. Authorisation automatism was realised in respect of this system as well, which means that every entrant or colleague shifting their position receives a profile appropriate for their newly fulfilled position.
- To provide support for the processes the workflow interface of the CTSA was introduced that traces even the authorisation changes of the non-managed systems because it was linked with the previous user administration system. Of course the workflow serves the satisfaction of other unique needs (withdrawal management, for example) as well, that is, it documents and supports the requesting, granting and deleting of any authorisation or profile.

5.3.4 Administration Related to the Introduction of the EIM System

Content of the administration part

In the following, we summarise briefly the two sections found in the description of the administration part. Following the introduction of the system the requisition of authorisation is realised mainly through the workflow interface. Accordingly, the ”Managing authorisations” part deals with the requisition and revocation of authorisations, and the “Managing the standard table” part contains the tasks related to the modification of the standard table.

Managing authorisations

In the case of the entry of a new colleague or the shift of assignments and tasks related to a position the new colleague must be assigned certain rights in the standard authorisation table. The assignment is performed according to the complex JobCode defined by the organisational unit and the assignment. Upon the entry of a new colleague or the transfer of an old one the data of the user are registered / modified in the HR system. Upon registration the user gains the basic rights (AD registration: email address, bank domain access) automatically with the help of the EIM system. At the same time, the Authorisation administrator receives an automated email notification about the enrolment. Having reconciled, in the next step, the complex authorisation profile (JobCode) with the employee’s supervisor the administrator sets the requested JobCode and starts the personalised authorisation request process in the Workflow. In the course of the authorisation request procedure, following an executive affirmation the user receives the requested authorisations automatically for the managed systems that run under the EIM. For the non-managed system the Remedy system notifies the teams of the solutions concerned to that perform the required and granted settings. In the case of a unique authorisation request a simple, managed systems’ JobCode request must be initiated in the Workflow. Once the requester registered the demand, the supervisors of the user have to approve of the request by answering an automatic email, then the same must be done by the Authorising group that inspect the request and the supervisory approval and then grant their own approval, where the authorisation process normally ends. If BankSecurity sees some obstructive factor concerning the granting of authorisation, it may refuse to grant the rights. In such a case it obligatorily notifies the supervisors approving the request. Following the approval of the request the user authorisations get set by the automatic workflow as described above. Regarding the deletion of a user’s rights, the superiors of the concerned employee have to decide, along with the Authorising group (unless the request came from human politics). Following the necessary decisions an automatic workflow is initiated during which the user’s authorisations are deleted in the managed systems and a Remedy case is set in motion in the case of the non-managed ones that the teams of the solutions concerned receive.

The deletion of the authorisations is initiated with the Authorisations administrator by:

a) **the supervisor of the user field:** for example in the case of a position transfer / shift

b) a member of Human Resources: for example in the case of a lay off**,**

c) **any user:** in the case of a shift in / cancellation of their own tasks, duties.

Managing the standard table

Modification of the standard table is necessary if

d) A new system gets incorporated into the central system of the EIM,

e) A system gets deleted from the central system of the EIM,

f) There is a shift in the authorisation settings / authorisation group of a system,

g) To map the authorisation of an organisational unit a new activity needs to be entered into the standard table or an old one must be deleted

h) A row belonging to an organisational unit must be modified in the standard table by adding or removing authorisations.

The listed options may, of course, apply at the same time as well. Upon the emergence of the need of modification the Standard Table Administrator reconciles the matter with everyone concerned (executives of specialties, system administrators), and modifies the standard administration table according to the jointly accepted results. Should development be necessary for the modification of the standard table, the Standard Table Administrator requests for quotation from the developer of the ESS and places the order for the development according to the "Investment, Cost Management and Acquisition Regulations".

The order of introduction progressed along the following points:

I. The first operative element of the system manifested in the HR / entry system – CTSA – AD/Exchange automatism.

II. Next the account management system went live, along with the CTSA workflow and the support of the administration as well.

Following the successful testing phase and the fruitful documentation, the evaluation of the pilot concluded that the system is able to satisfy the requirements of the bank and that the proposed objectives can be achieved, therefore in the second implementation phase all the additional bank applications must be brought under the CTSA "shield".

The second implementation phase is still being undertaken in the bank.

During this phase the data storage house, the commercial paper (stock) management system and the credit card system will become integrated.

During the live operation of the system the following roles (profiles) can be defined:

Tasks of the Authorisation administrator:

i. Ensuring the operation of the EIM environment.

ii. Fulfilment of business tasks related to the management of authorisation request.

iii. Maintenance of the application parameters necessary for the operation of the ESS system (administration of changes in the standard table into the ESS system, administration of organisational changes, cost location code changes and so on)

iv. Managing the access rights for the EIM system.

v. Maintenance of authorising groups.

vi. Verification of the service that sends in data regularly from the HR system.

vii. Regular verification of the EIM Audit.

viii. Investigation of the requests coming from BankSecurity, fulfilment of implementable requests

ix. Preparation of EIM reports.

x. Substitute for the Standard Table Administrator if that becomes necessary

Tasks of the Standard Table Administrator

i. Maintaining contact with the officers responsible for the applications of the managed systems.

ii. Observation of the changes in the organisational structure of the bank and their introduction into the EIM system (should the need arise, initiation of the modification of the standard table, for example).

iii. Analysis of the authorisation request patterns.

iv. Tasks related to the modification, adjustment of the standard table.

v. Substitution for the Authorisation administrator should the need arise.

ESS system administrator

vi. Fulfilment of the tasks related to the operation of the ESS system.

vii. Ensuring the availability of the ESS system

Authorising groups

viii. Authorising simple adjustments or deletion of authorisations in view of the relevant systems.

Officers responsible for given applications

ix. Initiation of a new system’s introduction into the EIM, initiation of the adjustment of a system’s authorisation

x. Initiation of the removal of an abolished system from the EIM.

5.3.5 Long Term Functional Expectations, Future Prospects

Single Sign-On

Control-SA provides password synchronisation for the supervised systems. What’s more, Control-SA/Agents can be interfaced with the leading Single Sign-On solutions, for example with the following: Symantec PassGo, CA eTrust SSO (previously Memco Proxima SSO), CyberSafe Trust Broker.

Public Key Infrastructure

BMC Software company is progressively working to combine the leading PKI standards and the PKI manufacturers' solutions with the Control-SA. The solution of Entrust PKI 5.0.x is already supported. Furthermore, the PKI functions of Microsoft Windows 2000 (including the SSO) are fully supported, and the full interface is ready for the RSA PKI solution of RSA Security as well.

Integrating the security systems

Every security system that has a user database that is accessible externally through some interface can be integrated into the CTSA system. This integration brings the given security under the EIM, however, linking the events of the security system with the EIM system is not completed simply by this. For example, the alerts of the entry system are independent of the alerts of the EIM system, no correlation can be set up between the two, even though the need for this could arise in several cases.

The CTSA/Links system is capable of receiving the alerts of various security systems even in such cases when the alert comes from a non-IT system (for example, when it appears on the screens or consoles of an entry or an observer system). The CONTROL-SA/Links can handle even these physical connections as well. With the help of the CTSA/Link system the events of security systems and the EIM system arrive at a common events server where they can be correlated, and where, if the proper user data is available in the security system, an immediate EIM operation can be initiated automatically.

Password synchronisation

The Control-SA system supports password synchronisation as well. This is an especially important function in an environment where several systems operate and where these systems have differently set or altogether different password generation and password utilisation regulations. Prior to password synchronisation it is useful therefore to harmonise the various password formation rules of the given systems. The synchronisation of the passwords of specific Enterprise Users must be authorised. This can be done in several ways, even during the entrance of the user. Upon password synchronisation, when a user changes their password within the RSS (resident security system) the agent software belonging to the RSS passes the new password on to the ESS which then attempts to set that for further RSS users too (who are linked to the Enterprise User). Password synchronisation can be observed even by the user, should they decide to change their password on the specific RSS systems not directly but through the dedicated web interface of the Control-SA/PassPort.

Authorisations dependent on security events

In connection with the chapter dealing with the intrusion detection concerning the EIM system and with the validation and exception management chapters of the Authorisation settings, Control-SA provides a way to define an endless number of authorisation adjustment rules for the security events discussed above. For example, a given security event could temporarily disable certain passwords, or restrict or expand the rights of certain users.

5.3.6 Summary and Assessment

The identity management system developed ensures access to the current state of bank system accesses for BankSecurity in a coordinated and transparent way. Internal Control is provided with both historical data on, and the features of authorisation changes, in the form of reports that can be utilised in target investigations. The HR specialty can directly initiate authorisation changes by entering or adjusting data concerning the enrolling, transferring or laying off of the employee in its own database. The IT specialty may free up resources by handing over the control of authorisation processes, and may remain in an executive role instead of a decision making one conforming to the original functional structure.

5.4 iManager – Identity Manager for Partial Identities

iManager is an identity manager for a mobile user in order to support him to communicate securely, to manage his partial identities, and consequently to protect his privacy.[78] This identity manager is a client side identity manager, which means that it is part of the user's mobile device, and can be classified as a type 3 IMS.

This work describes the term partial identity and the architecture of the *iManager*. A more detailed description and the use of *iManager* is described in [Ger03a] by an exemplary scenario: buying and controlling an electronic railway ticket.

5.4.1 Identity Management with Partial Identities

Every person has his own identity. This identity consists of person's roles, e.g. while using government services a person is well known whereas while he is shopping, he is almost anonymous. These changes of identity depending on the situation are represented by partial identities. A partial identity is a set of personal attributes of a user. A user can have several partial identities. Close to the physical world, a user changes his partial identity in computer networks while thereby varying between being anonymous and being fully identifiable. Such a change depends on the situation and role necessary for this situation. By this means, a user protects his privacy and at the same time is able to build up a reputation towards his communication partner with respect to his current role. The partial identity has been introduced by Roger Clarke in 1993, however not for privacy-enabling identity management, but for surveillance [Cla93]. The relationship between partial identities and authorisations by attribute certificates / credentials is described in [Cla01].

An example for using partial identities shows the following figure. The identity of the exemplary user called *Willi Weber* has four partial identities: anonymous, leisure, shopping, and public authority. By using a partial identity, he publishes some personal attributes, e.g. when using the partial identity *public authority* he publishes his name, birthday, place of birth, and his address. Whereas while using his partial identity *anonymous* he doe not publish any personal attribute at all. As a result, he is identifiable and he is able to establish a reputation with respect to the identity used while controlling the disclosure of his personal attributes and consequently protecting his privacy.

[78] The special characteristics of mobile identity management are described in FIDIS Deliverable 3.3 "Study on Mobile Identity Management".

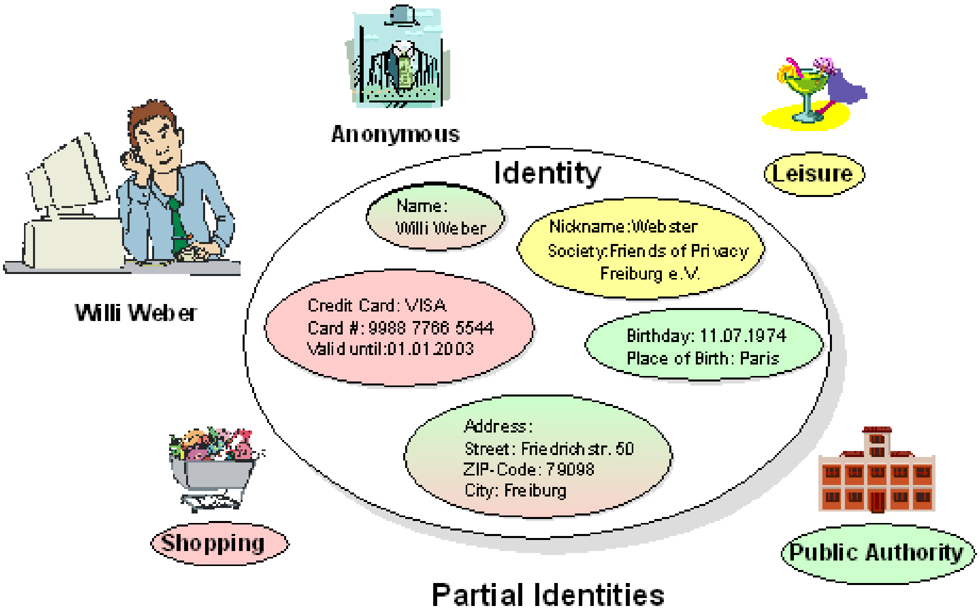


Figure 7: Identity and partial identities of an exemplary user

5.4.2 Data Structure of a Partial Identity

A partial identity is a record of personal attributes. Each record consists of a personal attribute and the corresponding personal data. A record is identified by a unique identifier.

- **Unique identitifier (pseudonym)**: A user is able to authenticate himself towards his communication partners with personal data, e.g. his name or with a cryptographic public key. Depending on the desired choice between being anonymous and fully identifiable, a user is able to use various kinds of pseudonyms [Pfi00]. Transaction pseudonyms, e.g. unambiguous transaction numbers, make possible the linkability of single steps within a transaction and to the user without revealing his identity. Whereas personal pseudonyms, e.g. telephone number, enables the personal identification of a user.
- **Identifier**: Each item of personal data is referenced by an identifier, e.g. “personname.given” (cf. [Cra02]).
- **Key identifiers** refer to an attribute which is unambiguous as for example a private cryptographic key.
- **Template**: A template of a partial identity is a set of identifiers which defines a partial identity but does not consist of any data and is used for creating a partial identity by a user.

Personal data and the settings of the user’s accountability that depends on a situation are stored by means of such a record for a partial identity. A user manages his partial identities with an identity manager. A research prototype for an identity manager of Freiburg University will be described in the following section. This identity manager called *iManager* enables a mobile user for managing his partial identities and thereby protecting his privacy.

5.4.3 Architecture of the iManager

The *iManager* is the central security tool of a mobile device which is considered to be trustworthy. The *iManager* offers interfaces to the user, to the security mechanisms, and to the applications of a mobile device. The access to personal data and to cryptographic keys is exclusively possible by using the identity manager. An application's request to these data will be checked by the identity manager whether the user has given his consent to the publication of this personal data in the current situation. The architecture of the *iManager* and its interfaces is shown in the following figure. Based on a *security platform*, the components *identity configuration*, *identity negotiation*, and *confirmation of action* are responsible for managing the partial identities [Jen01].

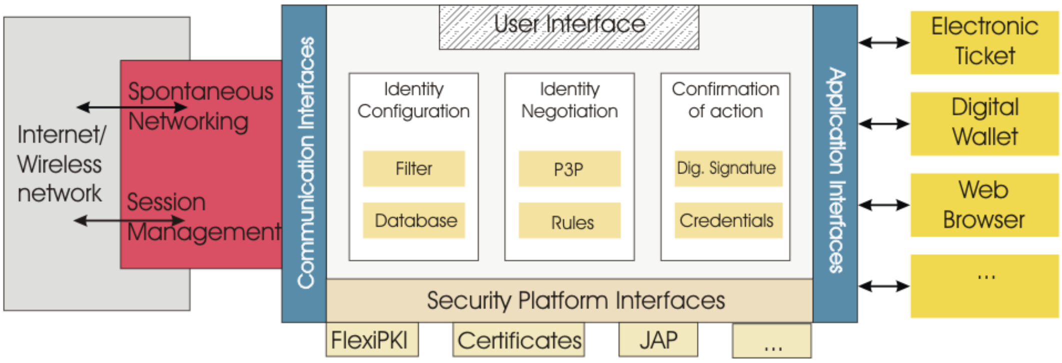
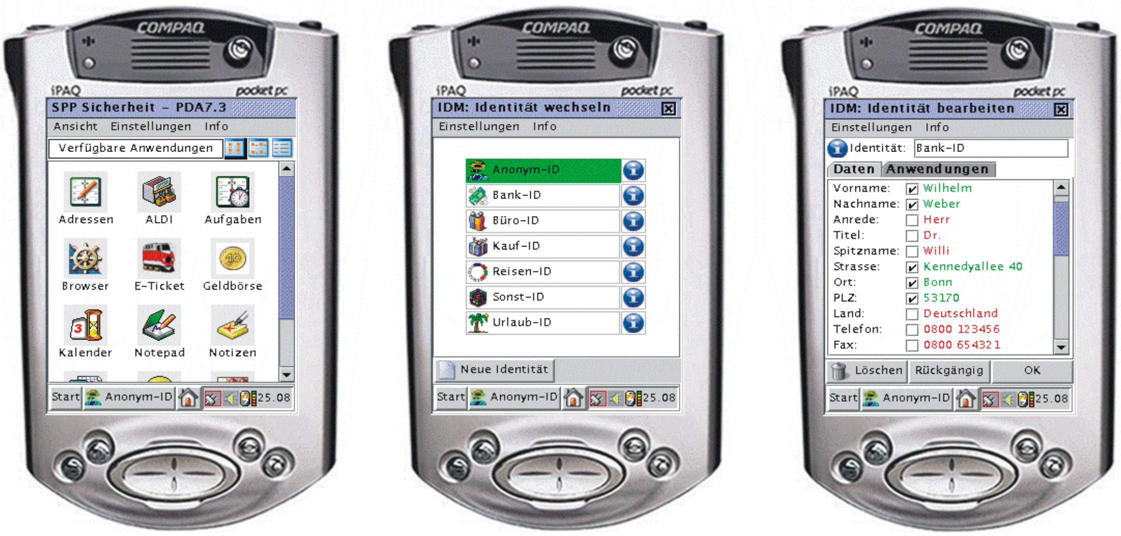


Figure 8: Architecture of the *iManager*

The *user interface* has to be comprehensible for security laymen, since they are not able to verify and assess the security mechanisms of the *iManager* and therefore a misuse of them leads to a compromise of the security and privacy of the user. The possibility of a misuse has to be reduced (cf. [Ger03b]). The acceptance of the security tool also depends on its user interface. In order to facilitate the use of a security tool, the protection goals of multilateral security [Ran97] have been classified in user and system controlled protection goals by analyzing their interdependency [Jen00]. This leads to a reduction of the user interface complexity. The user controlled protection goals *anonymity* and *accountability* are configured by partial identities and their choice in a situation. The integration of the *iManager* in the user interface of the mobile device is shown in the following figure. At any time, the user is able to check his identity.



Identität: Bank-ID

Vorname: ☑ Wilhelm
Nachname: ☑ Weber
Anrede: □ Herr
Titel: □ Dr.
Spitzname: □ Willi
Strasse: ☑ Kennedyallee 40
Ort: ☑ Bonn
PLZ: ☑ 53170
Land: □ Deutschland
Telefon: □ 0800 123456
Fax: □ 0800 654321

Figure 9: Integration of the *iManager* in the user interface of the mobile device

The *identity configuration* enables a user to choose and create a partial identity with respect to a current situation. A situation is defined by a communication partner, the current service and the current partial identity [Jen02]. Since the anonymity level cannot increase subsequently (monotony of anonymity [Wol00]) any partial identity can not be changeable. If the user wants to change the current partial identity, the *iManager* checks if the desired anonymity level could be reached with the intended change. Further implemented functionalities are: to edit partial identities, to store them in a secure database on the mobile device, and to recognise the current situation. The secure database stores partial identities and user's security, his privacy policies and rules for the security tools. A filter checks the data flow of the mobile device for personal data. By this means, it is possible to fill a web form according to P3P with respect to a suitable partial identity and user's permission.

An *identity negotiation* is necessary, if a service needs more data from the user than he wants to publish in this situation. This conflict can be solved with a negotiation between this service and the user. A restricted automatically negotiation is possible by the implementation of P3P and consequently the comparison from the service's and user's security and privacy policy. In case of a conflict, *iManager* informs the user of this conflict and proposes solutions like a suitable partial identity for solving it. For example, in the scenario a user wants to buy an electronic railway tickets and wants to get some premium points. For the premium points, the virtual ticket automat requests some personal data of the user. A conflict occurs since the user acts with his partial identity *anonymous*. The *iManager* proposes to use the partial identity *traveller* for solving this conflict. The following figure shows this case.

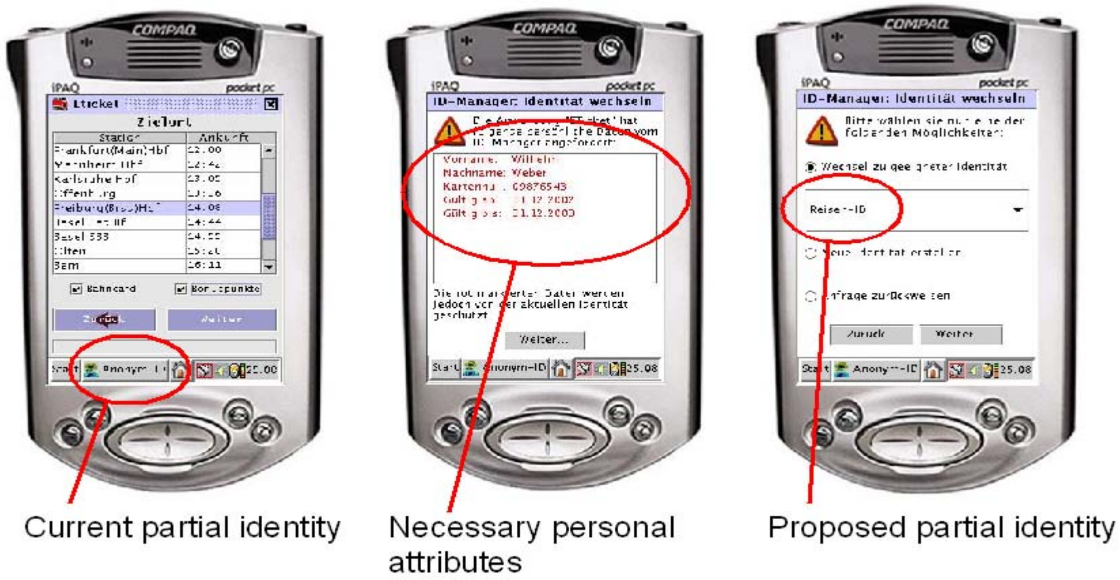


Figure 10: Identity negotiation

The user decides his accountability and the accountability of his communication partner for each partial identity. The component *confirmation of action* implements the accountability of the user by a digital signature tool. It is used whenever a digital signature is required, e.g. for self-signing personal data. Since the user declares explicitly his intent, he signs with his handwritten signature and authorises the digital signature tool to sign the corresponding credential. The digital signature key is selected by choosing the suitable partial identity. By this means, the technical functions of the key management will be shown in a more comprehensible manner [Ger01].

The *security platform* consists of interfaces to cryptographic primitives, anonymity services, to a session management, a secure database, and to security services. Anonymity services are the foundation of identity management, since it enables to user to be anonymous towards his communication partners. The anonymity service JAP [Ber00] is used for IP networks. For spontaneous networking, a library of Rostock University, Germany, [Sed01] is used. The cryptographic primitives for encryption and digital signatures are implemented by the library FlexiPKI [Buc99].

5.4.4 Summary

The *iManager* of Freiburg University, Germany, shows that it is feasible to realise privacy and security interests of a mobile user depending on the situation by managing and appearing with different partial identities. It is further developed in order to support privacy in business processes in which services are acting on behalf of the user and need access to user's profile which is stored by another service.

5.5 The idemix Credential System

Idemix is an identity management system based on anonymous credentials and zero-knowledge protocols. It was developed by the IBM Zurich Research Laboratory, Switzerland. In this system, users may maintain unlinkable pseudonyms with different organisations, obtain credentials signed by these organisations certifying certain attributes, and prove these attributes to verifying organisations.

By using *idemix*, users may have control on their identity attributes. They can choose which attributes they want to show or prove to a certain organisation. The system allows for minimal data leakage, as well as for pseudonymous identity management. Moreover, *idemix* implements accountability mechanisms, allowing for de-anonymisation under certain conditions.

This work describes the functionality of the *idemix* credential system. More details can be found in [Cam01, Cam02].

5.5.1 Basic Credential Protocols

The core of the *idemix* system consists of the protocols described in [Cam01]. This section describes these protocols in terms of parameterised primitives of which functionality can be easily explained and mapped to system interfaces.

The entities in the system are users, who obtain and show credentials, and organisations issuing and verifying credentials. Another type of organisation, de-anonymising organisation, is discussed below. Thus, a user U can obtain a credential C from an (issuing) organisation O_I; and then show the credential C to another (verifying) organisation O_V. A credential is always issued on a pseudonym N under which U is registered with (or known by) the issuing organisation O_I. A credential may have certain attributes (*attr*). When showing a credential, the user can choose which of the credential's attributes shall be revealed.

Pseudonym registration, credential issuing and credential verification are interactive protocols between the user and the specific organisation. A user U has a (single) master secret S_U, which is linked to all the pseudonyms and credentials issued to that user. Issuing and verifying organisations all have a public/private key pair (PK,SK). The organisation issuing a credential uses its private key to generate the credential; the credential can then be verified using the issuing organisation's public key, either by the user when receiving the credential, or later on by any organisation to which the user shows the credential. When showing a credential, the user uses the public key of the verifying organisation which, in turn, needs its private key in the protocol. Figure 5.5.1 shows the model for basic credentials protocols.

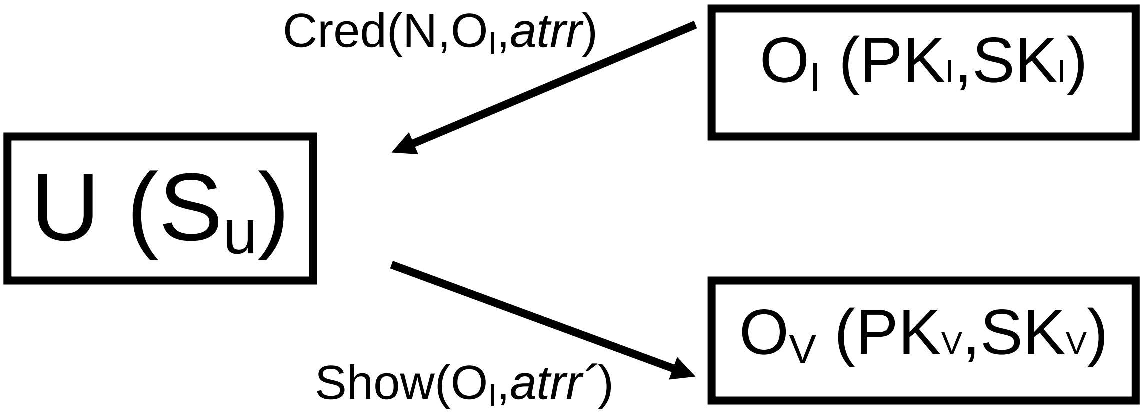


Figure 11: Basic credential system protocols

Obtaining a credential from O_I and showing it to O_V works as follows. First, U contacts O_I and establishes a pseudonym N with O_I. If N is eligible to get a credential with an attribute *attr*, O_I produces a credential C by signing a statement containing *attr* and N and sends C to U. Now U can to show this credential to O_V. That is, using a zero-knowledge proof, U convinces O_V of (1) possessing a signature generated by O_I on a statement containing *attr* and N, and (2) knowing the master secret key S_U related to N. We stress that U does not reveal any other information to O_V. In particular, U does not send O_V the actual credential. This way of showing a credential together with the zero-knowledge property of the proof ensures the unlinkability of different showings of a credential and also the unlinkability of a showing of a credential to the pseudonym to which the credential was issued. This means that U can show C to O_V (or any other verifier) an unlimited number of times, without these credential shows becoming linkable to each other or to a pseudonym. (Exceptions are one-show credentials, which are discussed in detail below). This unlinkability is maintained even if O_V and O_I are the same organisation (or pool their data).

Note that from this unlinkability property it follows that the user is anonymous towards the verifying organisation. Of course, this property of the pseudonym system can only provide real anonymity to the user if the communication channel used supports anonymity [Cha81].

While, in general, this approach to showing a credential is not very efficient, the special signature scheme used by the credential system [Cam01, Cam02] allows for an efficient realisation of the zero-knowledge proof described above. In fact, is indicated by our performance results, the computational complexity for both the user and the verifying organisation executing the protocol for showing a credential corresponds to generating a small number of signatures in the standard RSA signature scheme.

The fact that all of a user's credentials are linked to his master secret, sharing a credential would imply also giving away one's master secret not only ensure that user cannot pool their credentials but also allows to implement for measures to discourage users from sharing their credentials.

One way to do this is PKI-assured non-transferability, where the user's master secret key is tied to some valuable secret key from outside the system (e.g., the secret key that gives access to the user's bank account) [Dwo96,Gol98,Lys99]. Thus sharing a credential implies also sharing this valuable secret key. However, such a valuable key does not always exist. Another, novel way of achieving this is all-or-nothing non-transferability [Cam01]. Here, sharing just one pseudonym or credential implies sharing all of the user's other credentials and pseudonyms in the system, i.e., sharing all of the user's secret keys inside the system.

In cases where the verifier and the issuer is the same entity, the sharing credentials can be limited by the approach proposed by Stubblebine, Syverson, and Goldschlag [Stu99]. In this approach a credential can only be used once, but each time a credential is used, a new credential is issued. Thus, when a credential is given away, only the person using the credential first is given the next credential. This mechanism makes sharing access to a resource tedious.

Using the so-called Fiat-Shamir heuristic [Fia86], the protocol for showing a credential can also be turned into a signature scheme. The meaning of a signature will then be "a person possessing a credential issued by O_I has signed this message."

Both all-or-nothing non-transferability as well as the signature functionality will only be implemented in a future version of the prototype.

5.5.2 Credential Options and Attributes

Credentials can have options (such as one-show, or multi-show) and attributes. The one-show credentials incorporate an off-line double-spending test [Cha88]: when showing a one-show credential more than once (to the same or different organisations), this results in transcripts from which the issuing organisation can extract the pseudonym N on which it was issued.

Examples of credential attributes can be an expiry date, the user's age, a credential subtype. When showing a credential, the user can choose which attribute(s) to prove something about, and what to prove about them. E.g., when showing a credential that has attributes (exp-date = "2002-05-19", age = 55), the user can decide to prove only that age > 18.

5.5.3 Parameters of the Show Protocol

In this section, we describe two optional parameters that may be enabled when showing credentials. The first one allows for the implementation of accountability measures by requiring the user to provide encrypted identity information. The second one extends the functionality of the system by providing primitives with which the user can prove ownership of several credentials.

5.5.3.1 De-Anonymisible Show of a Credential

De-anonymising mechanisms allow to reveal the identity of a user (global de-anonymisation, also called global anonymity revocation) or to reveal a user's pseudonym with an issuing organisation (local de-anonymisation or local anonymity revocation). Global de-anonymisation allows for global accountability of transactions (e.g., for identifying a user performing illegal transactions); local anonymity revocation can be applied by the issuing organisation to take measures when a user misuses his credential. Both types of de-anonymisation are optional and require U's cooperation when showing a credential. They require the existence of a designated third party, a de-anonymising organisation O_D (see Figure 5.5.2) O_D has a public-private encryption-decryption key pair (PK_D, SK_D). Using this variant of the show protocol, U encrypts N with O_D's public encryption key. This encryption is verifiable (denoted $EV_D(N)$), which means that O_V has proof that O_D can decrypt and reveal the relevant N from O_V's show protocol transcript. There may be several de-anonymising organisations in the system, from which U may be able to choose. By including also a de-anonymisation condition, U can decide under which condition he consents to the transcript being de-anonymised. Later, when deemed necessary by O_V, O_V can send the transcript to O_D. O_D can then decide whether this condition is fulfilled and, if so, de-anonymise the transcript and returns N (local de-anonymisation).

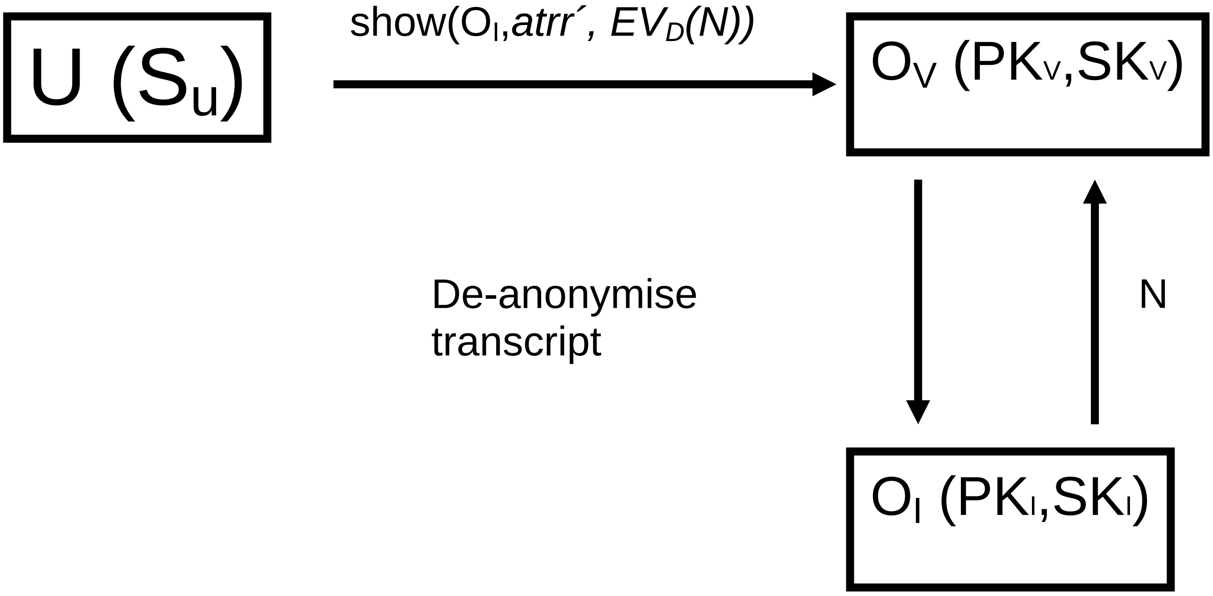


Figure 12: De-anonymisation

Global de-anonymisation uses essentially the same technique. It requires, in addition, the existence of a special credential issuing organisation, a Root Pseudonym Authority, which only issues credentials on pseudonyms of which it knows the mapping with a real user identity. A user typically has a single pseudonym (root pseudonym) with the Root Pseudonym Authority, and one credential (root credential) on that root pseudonym (additional pseudonyms or credentials with the Root Pseudonym Authority would anyway be linkable to the user).

5.5.3.2 Showing a Credential Relative to a Pseudonym

Using this option, U, who has obtained a credential C by O_I on N_I, and who is known under pseudonym N_V to O_V, proves possession of C to O_V, while also proving that the pseudonym to which C was issued belongs to the same user as does N_V. More precisely, the user proves that the same master secret key S_U that is linked to N_V is also linked to the credential C and the pseudonym (N_I) the credential C is issued on.

This option is mandatory for U to convince O_V of possession of several credentials. Without using the option, two users each possessing a different credential could each show their credential to O_V and fool O_V into believing that it talked to a single user possessing both credentials.

Furthermore, this option is also mandatory if showing of a credential is a precondition for a user to get another credential. The reason for this can be seen from the following example. Let us assume that U wants to obtain a credential from O_{VI}; O_{VI}, in order to issue such a credential, requires U to show a credential by O_I. If U has such a credential, he first registers a pseudonym N_{VI} with O_{VI}, and then shows the credential by O_I to O_{VI}, upon which O_{VI} considers the precondition to be satisfied and issues the new credential on N_{VI}. If U has no such credential, he can try to collaborate with U' (who does own the credential) by asking U' to perform the second step (showing the credential by O_{VI}). And indeed, if O_{VI} does not require U to show the O_I credential relative to a specific pseudonym, U will obtain the credential from O_{VI} without fulfilling the precondition. By requiring the show of the O_I credential relative to N_{VI}, O_{VI} enforces that the same user who showed the O_I credential gets the new O_{VI} credential.

5.5.4 Summary

Idemix implements a flexible user-controlled identity management system. Users have a master secret which is used in the protocols to securely establish pseudonyms with organisations; obtain signed credentials issued by the organisations; and show the credentials. The protocols do not leak any information on the identity of the user, as *idemix* implements “zero-knowledge protocols”. Note that an anonymous communication channel is required in order to protect the identity of the user at the communication layer.

Credential shows are unlinkable to each other and to the user’s pseudonym, even towards the issuing organisation. This allows for privacy enhancing identity management. *Idemix* implements optional accountability by requesting the user to provide a verifiable encryption of his pseudonym. Optionally, users may also prove ownership of several credentials.

6 Privacy Enhancing Concepts

Privacy enhancing concepts and the implementation of privacy enhancing technologies are important aspects of IMS. For IMS of type 3 they are unique selling propositions (USP) and so they are one factor deciding how successful an IMS can be on the market. In this chapter main conceptual mechanisms to support privacy are described and good practice examples of technical implementation (including some privacy enhancing technologies) are listed.

6.1 Mechanisms to Meet Requirements of IMS with Respect to Privacy

6.1.1 Introduction

The following categories and mechanisms are derived – among others – from [ICP03]. The categorisation is commented with respect to privacy.

The mechanisms are listed in sections. Each block has enumerations describing the mechanisms and listing partial mechanisms.

Section I to III is describing the main functionality of the different types of IMS.

Section IV is describing security as a mechanism. Generally security is also seen as fundamental for privacy.

Section V describes specific and general mechanisms to meet privacy requirements.

Section VI describes mechanisms to achieve interoperability. Depending on the type of IMS those requirements differ. Type 1 IMS especially require compliance with standards in the area of authentication systems (e.g. PKI) and directory services (e.g. LDAP, SAML etc.). Type 2 IMS require Interfaces to collect data generated or transferred from the user-client. Type 3 IMS require in addition to the standards mentioned for IMS Type 1 the integration of various existing identity management applications to larger identity management systems.

Section VII to X describes mechanisms for Type 3 IMS which are important to make the existing applications (today mainly client-side tools) attractive for the majority of users. Those mechanisms are meant to overcome the main observed obstacles for a better market penetration of Type 3 IMS.

6.1.2 Type 1 IMS: Account Management Systems

I. Functionality: Centralised and decentralised account administration
   a. Centralised creation of accounts, decentralised administration of identity information
   b. Centralised role Management
   c. Identity recovery

II. Functionality: Logging
   a. To determine the attempt to access restricted data

III. Functionality: Access control
   a. Authentication and application of roles
   b. Single Sign-On

IV. Security (the following aspects of Security are taken from[1], the IT Baseline Protection Manual[79] and the British Standards (ISO/EIC 17799)[80])
   a. Confidentiality (e.g. secrecy of authentication data)
   b. Integrity (including non-repudiation)
   c. Availability

V. Privacy
   a. Privacy control functionality (consent, objection, disclosure, correction, deletion and addition of privacy information)
   b. Role-based access to privacy information stored in the accounts
   c. Data minimisation: Storing and processing only data which is really necessary
   d. Standards (e.g. P3P[81]), seals (e.g. Privacy Seal by ICPP[82]) and penalties

VI. Interoperability for third party integration
   a. Compliance to existing standards
      i. Examples: LDAP, SAML etc.
   b. Well defined interfaces

6.1.3 Type 2 IMS: Profiling

I. Functionality: Logging user interaction and generate profiles for further internal use

[79] See: http://www.bsi.de
[80] E.g., http://www.iso17799-web.com
[81] http://www.w3.org/P3P/
[82] http://www.datenschutzzentrum.de/guetesiegel/

II. Functionality: Notice
   a. Share profile data with the user

III. Functionality: User control
   a. Rule handling
      i. User is in control of the data transferred into the profile or the profile itself (by local storage and central processing)

IV. Security
   a. Confidentiality (e.g. anonymisation, secrecy)
   b. Integrity (including non-repudiation)
   c. Availability

V. Privacy
   a. Privacy control functionality (consent, objection, disclosure, correction, deletion and addition of privacy information)
   b. Data minimisation: Storing and process only data which is really necessary
   c. Standards (e.g. P3P), seals (e.g. Privacy Seal by ICPP) and penalties

VI. Interoperability for third party integration
   a. Compliance to existing standards
   b. Well defined interfaces

6.1.4 Type 3 IMS: User-Controlled Context-Related Role and Pseudonym Management

I. Functionality: Identity administration
   a. Communication-independent handling and representation of identities: Possibility to choose between different profiles / data schemes; creating, updating, deleting identity and identity information
   b. Pseudonyms with specific properties: Using pseudonyms for privacy enhancement by averting linkability
   c. Credentials: To reach an optimised privacy protection credentials can be used as convertible certifications by which authorisations obtained under one pseudonym can be transferred to another pseudonym without loosing unlinkability. Although an authorisation is bound to an individual and can be reliably used in many contexts, its use does not automatically lead to data trails or unwanted disclosure of personal data. As long as the individual does not misuse the credential, anonymity is guaranteed.
   d. Identity recovery: A user may want to prove that a given pseudonym was in his control at an earlier time.

II. Functionality: Notice
   a. History management: Possibility to log transaction for reconstructing and analysing data flow
      i. Example: Illustrating what the communication partner knows from previous transactions; filters could be used to get a view e.g. on identity and identity information
      ii. Practical view: email communications have to be stored completely, because the privacy-relevant content cannot be analysed automatically.
   b. Context detection: which partial identity was used in which transactional context?

III. Functionality: Control
   a. Rule handling
      i. Support user to choose the right profile / preferences etc.
   b. Anonymity as base-rule for privacy enhancement
      i. Essential on the lower layers to enable Identity Management
      ii. Anonymity is also seen as mechanism for security, especially confidentiality

IV. Security
   a. Confidentiality (e.g. anonymity, secrecy)
   b. Integrity (including non-repudiation)
   c. Availability (e.g. if a cascade within anonymising service such as JAP/AN.ON goes down an automatic redirect to another cascade takes place)

V. Privacy
   a. Privacy control functionality (consent, objection, disclosure, correction, deletion and addition of privacy information)
   b. Data minimisation: Storing and process only data which is really necessary
   c. Standards (e.g. P3P), seals (e.g. Privacy Seal by ICPP) and penalties

VI. Interoperability for third party integration
   a. Compliance to existing standards
   b. Well defined interfaces for integration in popular software (e.g. mailers, browsers, etc)

VII. Trustworthiness
   a. Segregation of power, separating knowledge, reviewing by independent parties
   b. Using Open Source
   c. Trusted seals of approval

VIII. Law Enforcement / Liability
   a. Digital evidence
      i. Example: Proof of transactions etc.
   b. Digital signatures

c. Data retention
   i. Comment: this is contrary to privacy

IX. Usability
   a. Comfortable and informative user interfaces
   b. Training and education
   c. Low complexity
   d. Raising awareness

X. Affordability
   a. Power of market: Create IMS that are competitive and are able to reach a remarkable penetration of market
   b. Using open source building blocks
   c. Subsidies for development, use, operation, etc.

6.2 Good Practice Examples and Considerations of Privacy Enhancing Technical Implementation of These Mechanisms

6.2.1 Type 1 IMS: Account Management Systems

Personal data about employees and customers is often stored in so-called directory services (e.g. NDS, MS ActiveDirectory, LDAP, X.500). From a privacy perspective, it would be desirable if different groups or departments (or roles) would have different views on subsets of the personal data. For example, the authentication server should have access to authentication data of an employee, but not to address, phone number, etc. This can be achieved by role-based access control.

6.2.2 Type 2 IMS: Profiling Systems

Type 2 IMS will be subject of Deliverable 7.2 (Inventory on actual profiling techniques and practices). In this chapter we discuss some technical and organisational privacy enhancing techniques which are quite apparent. In some cases they collide with the targets of the operators of the profiling systems. To get a more practice oriented view on those techniques the scenarios in Chapter 2 of Deliverable 2.2 (Set of use cases and scenarios) can be used.

Section I - III (Functionality):
Profile data could be stored local and processed centrally.

Section IV (Security)
Data transfer is appropriately encrypted.

Section V (Privacy) and section VI (Interoperability):

The operator of the profiling system has a privacy rsp. data protection policy. Transferred data is negotiated automatically based upon privacy principles stated by the user using protocols like P3P. The transfer of the profile data needs the explicit consent of the user.

6.2.3 Type 3 IMS: User-Controlled Context-Dependent Role and Pseudonym Management

We list examples of e-mail, web and instant messaging software and configurations which partially implement the mechanisms categorised above.

Section I (Functionality: Identity administration):
Management of identities: Many mail readers facilitate automated handling of multiple e-mail accounts; to name a few: Eudora[83], Mozilla Thunderbird[84,] fetchmail[85]. Several web browsers have so called password managers to store username/password pairs per context; e.g., Mozilla[55], Opera[56]. Divers instant messaging clients manage multiple nicknames for the user; e.g., gaim[59], irssi[86].
Credentials: Although the technique is published (see *idemix*, Chapter 5.4), there seems to be no available implementation of credentials for either e-mail, http or instant messaging.
Identity Recovery: We found no example.

Section II (Functionality: Notice):
Logging/history: Practically all e-mail readers allow logging of sent/received messages. Some instant-messaging clients allow logging of conversations. Local caching HTTP proxies like wwwoffle[87] allow inspection of downloaded data to get an idea of which servers were connected to. There seems to be no convenient way of evaluating the logs automatically, though.
Context detection: Several web-browsers feature automated form-filling and password management as mentioned above.

Section III (Functionality: Control):
Rule handling: Internet Explorer[57] implements (at least partially) P3P.
Anonymity: JAP[88] helps to anonymise HTTP traffic, Remailers like mixmaster[37] or mixminion[38] to anonymise e-mail, services like Tor[89] facilitate anonymity of TCP connections in general.

[83] http://www.eudora.com/

[84] http://www.thunderbird-mail.de/

[85] http://www.catb.org/~esr/fetchmail/

[86] http://www.irssi.org/

[87] http://www.gedanken.demon.co.uk/wwwoffle/

[88] http://www.anon-online.de/

[89] http://tor.eff.org/

Section IV (Security):
Confidentiality: The OpenPGP and S/MIME[90] standards define e-mail encryption and authentication. Tools like PGP[43], gnupg[44] and openssl[46] implement the standards. SSL/TLS for HTTP encryption is implemented in most web-browsers, e.g, Links, Lynx[91], Opera[56], Mozilla Firefox[92]. A few instant-messaging clients allow encryption, e.g., skype[93], jabber[58], gaim[59] with patches, silc[94], Trillian[95].
Integrity: Comes as side-effect of strong encryption.
Availability: Applications depend on many layers and components. Methods for assurance of availability range from redundant hardware to remembering to pay for the DNS entry.

Section V (Privacy):
Privacy control functionality and data minimisation are not easy to apply to client-side technologies. One example would be HTML forms that mark certain fields as optional, or web pages that present a privacy policy.
Standards: The TRUSTe[96] seal announces conformance to TRUSTe's License Agreements. These include among other points data minimisation. However, on one occasion[97] of blatant privacy policy violations, TRUSTe proved extremely reluctant to revoke their seal on the offending site.

Section VI (Interoperability):
Compliance to existing standards: as most tools try to implement identity management aspects on top of e-mail, HTTP or existing instant-messaging protocols, the tools have to comply to the standards that define the underlying protocols.
Interfaces: Strict definition of the protocols behind the protocol used in Tor[89] allowed an alternative, interoperable implementation.

Section VII (Trustworthiness):
Segregation of power, separating knowledge, reviewed by independent parties:
The businesses of authoring software and providing services are so different that the involved parties do not overlap.
Using Open Source: Most of the tools of type 3 IMS run on the client side or use peer-to-peer protocols, an area full of open source implementations.
Trusted seals of approval: There seems to be no example.

[90] http://www.imc.org/smime-pgpmime.html
[91] http://lynx.browser.org/
[92] http://www.mozilla.org/products/firefox/
[93] http://www.skype.com/
[94] http://www.silcnet.org/
[95] http://www.trillian-messenger.de/
[96] http://www.truste.org/
[97]http://dir.salon.com/tech/log/1999/11/09/truste/

Section VIII (Law Enforcement / Liability):
Digital evidence and digital signatures: Existing tools implement “advanced signatures” according to the European signature framework[98] Using, e.g., Mozilla’s PKCS#11 plug-in together with a smartcard could meet the requirements for even qualified signatures.
Data retention: Since end-users are not required by law to keep any transaction logs, there is no example.

Section IX (Usability):
Comfortable and informative user interfaces: The variance is extremely high. Some tools like the HTTP proxy privoxy[12] show useful information and are easy to configure, while, e.g., the old PGP-based remailer network is extremely hard to use.
Training and education: Given the non-commercial nature of many of the tools there is almost no training available.
Low complexity: this varies greatly, depending on the problem. The complexity is comparatively low in HTTP-proxies like junkbuster[99], privoxy[12], webwasher[100], and quite high in mail-processing tools like mixmaster[37].
Raising awareness: the grass-roots nature of the market precludes systematic marketing attempts. Privacy problems are pointed out mainly by journalists, hackers and privacy protection officials.

Section X (Affordability):
Power of market: E-mail and web depend strongly on interoperability and conformance to the relevant IETF standards. As these are open standards, there is a healthy competition between implementers of clients or servers in this area. Commercial providers of instant-messaging repeatedly tried to stifle competition by changing their protocols between versions of their client software. In 2004, the IETF decided to standardise an instant-messaging protocol along the lines of an existing non-commercial one (Jabber[58]). It remains to be seen if this will create stronger competition. There are open source implementations of every protocol for e-mail (in fact, the majority of mail transport agents that provide the e-mail infrastructure are open source) and HTTP (in fact, the majority of web servers are open source). Government subsidies are rare, a notable exception being projects developed and maintained at state-owned universities.

6.3 Summary

The privacy enhancing mechanisms shown in this chapter are homogeneous in their structure of sub mechanisms, but differ due to different tasks of identity management in content. For type 1 and 2 IMS these mechanisms can be implemented with the already used central systems depending on the purpose for which they are used. In any case a planning phase allowing for privacy compliant concepts is necessary.

[98] Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures; see http://europa.eu.int/eur-lex/en/dat/2000/l_013/l_01320000119en00120020.pdf

[99] http://internet.junkbuster.com/

[100] http://www.webwasher.com/

For type 3 IMS we find a large variety of different technical implementations of the described mechanisms for different internet services. A characteristic of those technical implementations is the lack of integration even for a single internet service like e.g. e-mail. This results in poor usability for the user trying to achieve privacy through identity management.

7 Research and Development Topics

As type 2 IMS (profiling systems) are handled in a separate Deliverable of FIDIS, this chapter concentrates on recent research and development concerning type 1 and type 3 IMS.

7.1 Research and Development in Type 1 IMS

According to current studies carried out e.g. by the Gartner Group, the Yankee Group[102] and the Radicati Group[101], the market of IMS is expected to grow fast at least until 2008. Turnaround prognosis starts from 748 million US $ in 2004[101] and varies from 3 billion US $ in 2007[102] to 10.2 billion US $ in 2008[101]. The expected turnaround is for IMS of type 1.

In general we observe two technical directions of development of IMS type 1:

1. Account management systems are integrating more and more in integrated solutions like enterprise resource planning (ERP). In addition to the described integration into human resource planning systems further integrations can be observed, e.g., customer relationship management systems (CRM)[103].
2. Integration of more organisations while distributing the administration:
   To distribute the account administration concerning some personal data to different administrators, federated identity management systems[104] can be used. With federated identity management systems and meta-directories (e.g. DirX by Siemens) implementations to support authentication, authorisation and accounting in groups of trusted organisations are becoming more and more available.

Scientific research concerning type 1 IMS currently is directed towards further development of advanced, centralised authentication and authorisation services. One example is the "Configurable Internet Directory and Authentication Service" (CIDAS)[105] developed by F.-L. Holl et al. Basing on an LDAP server, CIDAS offers among others different levels of authorisation depending on the chosen type of authentication and anonymous authentication against an application server. Currently an early prototype of the planned Open Source implementation is available.

7.2 Research and Development in Type 3 IMS

Designers of decentralised IMS (Type 3) try to give users control over their personal data. Ideally, the data should be kept on a machine under the user's control or under the control of

[101] See http://www.radicati.com/pubs/news/Q3-2004_PressRelease.pdf
[102] Cited by http://www.informationweek.com/story/showArticle.jhtml?articleID=18312163
[103] See http://de.sun.com/homepage/feature/2004/identity_management/
[104] See http://news.zdnet.com/2100-1009_22-5535345.html
[105] See http://www.cidas.org

someone closer to and more trusted by the user than a central authority. This is e.g. in compliance with the “Laws of Identity” put forth by Kim Cameron[106] of Microsoft, Inc. The following projects try to implement these requirements:

7.2.1 Identity Commons / OASIS XDI/XRI Standards

This technological approach[107] is still in the planning stage. The goal is to formulate access rules to personal data in a XML-defined form and store personal data together with the user’s rules about disclosure and processing. These data/rules objects are kept by “i-brokers” who make transactions for the users, always in accordance with the rules stated by the owners of the data. The brokers are expected to check user’s real-world identities to encourage trust in the corresponding virtual identities. The objects are identified by “i-names” and “i-numbers”, the latter are fixed and globally unique.

7.2.2 LID - Light-Weight Identity Management

This approach[108] uses URLs as identifiers. They are not bound to persons in any way. There is no central authorised party or layer of authorised parties. Behind each URL is a CGI which accesses the data available. The scheme uses HTTP mechanisms to enable such scripts to communicate with each other. They can then authenticate and negotiate which information is transferred. As a side effect, the system can be used for Single Sign-On.

A quite similar project is Sxip[109].

What distinguishes the schemes from directory-lookup services is the use of “smart pointers”. The identifiers cannot be de-referenced without calling a routine under the control of the user (in LID) or of someone trusted by the user (in Identity Commons). A smart pointer can run checks on the requesting party before de-referencing the data it points to.

7.2.3 Biometric Secured Client-Side Identity Management

This is an additional approach[110] to enable an improved management of biometric authentication data by the user himself. The basic idea behind this concept is the storage of a biometric template on a device controlled by the user and the use of biometrics and knowledge on that device for authentication purposes. The device generates in case of positive authentication a digital credential which can be used for authentication purposes with

[106] See http://identityblog.com/

[107] See http://idcommons.net

[108] See http://lid.netmesh.org

[109] See http://www.sxip.com

[110] See http://www.axsionics.ch/

various applications. This concept will be further described in the FIDIS study "PKI and Biometrics". Today an advanced prototype of this approach is available.

7.2.4 Integration, Advanced Prototyping and Basic Research

Basing on scientific research and development carried out by a consortium comprising enterprises, universities and a privacy commissioner, the project "Privacy and Identity Management for Europe"[111] (PRIME) currently aims at developing an advanced prototype for type 3 IMS. In contrast to prior research and prototypes the integration of a number of various identity management functionalities in one application is planned. In the area of research improvement of identity management functionalities, integration of credential systems, policy management, improved usability and user interfaces, business models and social acceptance are central topics of research and development within this project. PRIME is addressing policy-makers, business, administration, academia and standardisation organisations.

Many other groups are conducting research in the field of privay enhancing identity management, too. There is a huge variety of open research topics, e.g. improvement of anonymity or unlinkability, measuring privacy aspects and giving reasonable feedback to the user, communicating legal rights to the user and enabling him to really use these rights, privacy management languages and protocols, or trusted systems and their control. A further challenge is the integration of legacy systems, especially in the area of (e-)government area where the user's identity is increasingly represented by digital ID cards. The integration of (privacy enhancing) identity management systems into real world application ranges from small pseudonymisation concepts to systems which supports the user's privacy against increasing surveillance and decreasing transparency about what is happening with the user's data. Especially the way to an ambient intelligence society needs to be evaluated and designed with respect to privacy and identity management.

7.3 Summary

For IMS of type 1 according to current studies we observe a rapidly growing market and a tendency for proceeding technical integration of neighboured applications.

For IMS of type 3 we observe - next to the development of credential systems described in Chapter 5.5 – the development of new standards for access rule management and the smart use of URLs as de-referenceable identifiers. There are many open research issues with respect to type 3 IMS covering not only a technical approach, but also socio-economics, law, or ergonomics.

[111] See http://www.prime-project.eu.org/

8 Conclusion and Outlook

In general, we notice that the originally quite strict borders between the defined three types of IMS are diminishing. Type 1 IMS (account management systems) currently are expanding towards customer relationship management (CRM), which could as well be used in the context of type 2 IMS. In addition to the organisation-side view type 2 IMS (profiling systems) have a client-side view, which could as well be considered to be identity management of type 3. The categorisation into three types originally designed for different products still serves well to describe a certain view on more and more integrated solutions.

To demonstrate further areas of activity within the FIDIS Network we want to use a common economic model [SCH96]. This describes four phases for the lifecycles of products:

1. Phase 1: experimental or access phase
2. Phase 2: expansion phase
3. Phase 3: maturity and stagnation phase
4. Phase 4: regression phase

Using this model with the identified types of IMS we observe that IMS of the types 1 and 2 are in the second phase (expansion) of this model, IMS of type 1. The mechanisms of market (like competition of various manufacturers, supported standards like LDAP, SQL etc.) are working quite well with these types of IMS. Looking at IMS of type 3 we observe that they are in the phase 1 (experimental). The large variety of existing solutions presented in this document, the low degree of commercial activities (compared to the IMS of Type 1 and 2) and significant public activities (public promoted projects, public research) lend support to this classification.

Looking at technological aspects of the described types of IMS there is no public technology promotion necessary for IMS of type 1 and 2. Areas of research and development are integration of related and so far independent systems and technologies. This could lead to further development of the framework of European legislation (especially in the sector of privacy compliance) or its application.

While the necessity for activities in the legislation is the same with type 3 IMS as with IMS of type 1 and 2 there are additional needs. Barriers towards expanding markets (phase 2 of the economic model) and possible activities for overcoming those barriers are:

- The perception differs widely of what identity management is. A clearer taxonomy and public awareness are necessary.
- While current concepts and technologies for identity management are not commonly understood new technologies such as RFID and Ambient Intelligence are emerging. The technical opportunity of remote readout of e.g. the RFID without any notice by the user raises new questions towards identity management. Most today established IMS know an authentication done actively by the user.
  In addition known technologies, such as the use of mobile devices and biometrics are developing towards new services or applications (e.g. location based services and ID documents). The public reception influenced by technology friendly placement and a lack of integrated concepts is dominated by the discussion of risks. Technological,

political, social and economic opportunities have to be looked at in combination with legislation (including human rights and privacy compliance). As a result there will be recommendations for further integrated technological development and development of legislation towards those technologies.

- Integration of the existing, technologically feasible solutions is generally poor, interoperability therefore a major area of interest.
- While there are some prototypes with good usability features (e.g. *iManager*), many tools and application examined in this document are of poor usability (e.g. first generation remailers). This applies especially to those tools addressing special technical solutions for privacy compliance. To gain a better acceptance in the market, usability has to be improved.
- For type 3 IMS privacy, compliance is a unique selling proposition. On the other hand dependability and risk minimisation (understood as elements of security) are important for the provider of commercial or governmental services. This disjunction is leading to a separate discussion on fraudulent use together with criminal and forensic aspects of identity and identity management. Recommendations for further development of legislation based on an integrated understanding of the underlying technologies and social systems could be one result of this discussion.

The FIDIS Network of Excellence will address those aspects. The workpackages in the current workplan are designed to meet the described requirements. Basing upon the results and suggestions of this and other documents produced within the FIDIS Network the workplan will be updated.

9 Glossary

- **Anonymity**

  Anonymity is the state of being not identifiable within a set of subjects, the anonymity set.

- **Authentication**

  Authentication in the context of Identity Management is the process of validating the alleged identity of a person. Authentication requires that a user (intending to perform a specific action) provides a credential that proves he is in fact the person he claims to be.

  There are three main categories of credentials used to authenticate human users. Credentials are based on

  - something the user is (identifiers based on biometrics);
  - something the user has (hardware-based tokens such as smart cards/ software tokens such as digital certificates);
  - something the user knows (password or PIN).

  Authentication methods can be combined in order to increase accuracy. It should be noted that while authentication is usually based on identification, authentication without identification is possible as well.

- **Authorisation**

  The process of determining, by evaluating applicable access control information, whether a user is allowed to have the specified types of access to a particular resource is called authorisation. This always requires authentication. Once a user is authenticated, he may be authorised to perform different types of access.

- **Credential**

  In the widest sense, a credential is a piece of information attesting to the truth of certain stated facts. Credentials are used in the process of authentication, and in this context are based on the following technologies: Biometrics, digital certificates, smart cards, passwords etc.

- **Identification**

  Identification of a subject is the process of linking this subject to an identity.

- **Identifier, ID**

  An identifier (ID) is a name or string of bits. IDs can be assigned to subjects and objects. An identifier for a subject – with respect to a given community – is any information that uniquely characterises this subject in this community.

- **Identity**

  An identity is a set of characteristics representing a subject.

- **Identity management**

  Identity management means managing the various partial identities, i.e., their valuation as "applicable to one self" (role taking) or forming them (role making). A prerequisite to choose the appropriate partial identity is to recognise the situation the person is acting in.

- **LDAP – Lightweight Directory Access Protocol**

  LDAP is a directory access protocol standardised by the Internet Engineering Task Force (IETF) within the Requests for Comments (RFCs) 1777, 1778, 1779 and 1781. It describes the communication between directory clients and servers and the structure of the content of directories, not the content of them itself.

- **Partial Identity**

  Each identity of a subject can comprise many partial identities of which each represents the subject in a specific context or role. Partial identities are subsets of attributes of a complete identity. On a technical level, these attributes are data.

- **Privacy**

  Privacy is the ability of a person to control the availability of information about and exposure of himself or herself. It is related to being able to function in society anonymously (including pseudonymous or blind credential identification).

- **Unlinkability**

  Unlinkability of two or more items (e.g., subjects, messages, events, actions, ...) means that within this system, these items are no more and no less related than they are related concerning the a-priori knowledge.

- **XML – Extensible Markup Language**

  XML standardised by the World Wide Web Consortium (W3C) is a simple, flexible text format derived from SGML (ISO 8879).

10 References

[Ber00] O. Berthold, H. Federrath, and M. Köhntopp. Project "Anonymity and Unobservability in the Internet". In *Workshop on Freedom and Privacy by Design / Conference on Freedom and Privacy 2000*, pp. 57-65, Toronto/Canada, April 2000.

[Buc99] J. Buchmann, M. Ruppert, and M. Tak. FlexiPKI - Realisierung einer flexiblen Public-Key-Infrastruktur. *Technischer Bericht der TU Darmstadt,* December 1999.

[Cam01] J. Camenisch and A. Lysyanskaya. Efficient non-transferable anonymous multi-show credential system with optional anonymity revocation. In *Advances in Cryptology – EUROCRYPT* 2001, vol. 2045 of LNCS, pp. 93-118. Springer Verlag, 2001.

[Cam02] J. Camenisch and E. Van Herreweghen. Design and Implementation of the idemix anonymous credential system. In *Proceedings of 9th ACM Conference on Computer and Communications Security*. ACM Press, 2002.

[Cha81] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. In *Communications of the ACM*, 24(2): pp. 84-88, Feb. 1981.

[Cha88] D. Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In *Advances in Cryptology -- CRYPTO* '88, vol. 403 of LNCS, pp. 319-327. Springer Verlag, 1990.

[Cla93] R. Clarke. Computer Matching and Digital Identity. In *Proceedings of the Computers, Freedom & Privacy Conference*, San Francisco, 1993.

[Cla01] S. Clauß and M. Köhntopp. Identity management and its support of multilateral security. *Computer Networks*, 37(2): pp. 205-219, October 2001.

[Cra02] L. Cranor, M. Langheinrich, M. Massimo, M. Presler-Marshall, and J. Reagle. The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. http://www.w3.org/TR/P3P, April 2002.

[Dwo96] C. Dwork, J. Lotspiech, and M. Naor. Digital signets: Self-enforcing protection of digital information. In *Proceedings of 28th Annual ACM Symposium on Theory of Computing* (STOC), 1996.

[Fia86] A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Advances in Cryptology -- CRYPTO* '86, vol. 263 of LNCS, pp. 186-194. Springer Verlag, 1987.

[Ger01] D. Gerd tom Markotten, U. Jendricke, and G. Müller. Benutzbare Sicherheit – Der Identitätsmanager als universelles Sicherheitswerkzeug. In G. Müller und M. Reichenbach (Eds.), *Sicherheitskonzepte für das Internet*, Kapitel 7, pp. 135-146. Springer-Verlag Berlin, Mai 2001.

[Ger03a] D. Gerd tom Markotten, S. Wohlgemuth, and G. Müller. Mit Sicherheit zukunftsfähig. *PIK Sonderheft Sicherheit* 2003, 26(1): pp. 5-14, 2003

[Ger03b] D. Gerd tom Markotten. Benutzbare Sicherheit für informationstechnische Systeme, *Dissertation* at Albert-Ludwigs-Universität Freiburg, 2003.

[Gol98] O. Goldreich, B. Pfitzmann, and R. Rivest. Self-delegation with controlled propagation - or - what if you lose your laptop. In *Advances in Cryptology - CRYPTO* '98, vol. 1642 of LNCS, pp. 153-168. Springer Verlag, 1998.

[ICP03] ICPP (2003). Independent Centre for Privacy Protection (ICPP) Schleswig-Holstein and Studio Notarile Genghini (SNG); *Identity Management Systems (IMS): Identification and Comparison*; study prepared under contract for Institute for Prospective Technological Studies, Joint Research Centre Seville, Spain, Sept. 2003; http://www.datenschutzzentrum.de/idmanage/study/ICPP_SNG_IMS-Study.pdf

[Jen00] U. Jendricke and D. Gerd tom Markotten. Usability meets Security - The Identity-Manager as your Personal Security Assistant for the Internet. In *Proceedings of the 16th Annual Computer Security Applications Conference*, pp. 344-353, December 2000.

[Jen01] U. Jendricke and D. Gerd tom Markotten. Identitätsmanagement: Einheiten und Systemarchitektur. In D. Fox, M. Köhntopp, and A. Pfitzmann (Eds.), *Verlässliche IT-Systeme - Sicherheit in komplexen Infrastrukturen*, pp. 77-85. Vieweg, Wiesbaden, September 2001.

[Lys99] A. Lysyanskaya, R. Rivest, A. Sahai, and S. Wolf. Pseudonym systems. In *Proceedings of Selected Areas in Cryptography*, vol. 1758 of LNCS. Springer Verlag, 1999.

[Pfi00] B. Pfitzmann, M. Waidner and A. Pfitzmann. Secure and Anonymous Electronic Commerce: Providing Legal Certainty in Open Digital Systems Without Compromising Anonymity. *Technical Report* RZ 3232 (93278) 05/22/00, IBM Research Division, Zürich, Mai 2000.

[Ran97] K. Rannenberg, A. Pfitzmann and G. Müller. Sicherheit, insbesondere mehrseitige IT-Sicherheit. In G. Müller and A. Pfitzmann (Eds.), *Mehrseitige Sicherheit in der Kommunikationstechnik*, pp. 21-29. Addison-Wesley Longman Verlag GmbH, 1997.

[Sch96] H. Schmalen, *Grundlagen und Probleme der Betriebswirtschaft*, pp. 492-494, Wirtschaftsverlage Bachem, 1996

[Sed01] I. Sedov, M. Haase, C. Cap, and D. Timmermann. Hardware Security Concept for Spontaneous Network Integration of Mobile Devices. In *Proceedings of the International Workshop "Innovative Internet Computing Systems"*, Ilmenau, Juni 2001.

[Stu99] S. G. Stubblebine, P. F. Syverson, and D. M. Goldschlag. Unlinkable serial transactions: Protocols and applications. In *ACM Transactions on Information and System Security*, 2(4): pp. 354-389, Nov. 1999.

[Sun05] Sun Java™ System Access Manager 6. Technical Overview 2005Q1. Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054, U.S.A., Part No: 817-7643-10.

[Wol00] G. Wolf. and A. Pfitzmann. Properties of protection goals and their integration into a user interface. *Computer Networks*, 32: pp.685-699, 2000.

11 Indices

11.1 Index of Figures

Figure 1: Components of the identity platform 27
Figure 2: A Cross-Domain SSO authentication process 29
Figure 3: Overview of different web services specifications 32
Figure 4: WS-Trust/WS-Federation example 35
Figure 5: Overview of participants of the EIM software market 40
Figure 6: Universal control-SA architecture of the planned solution 42
Figure 7: Identity and partial identities of an exemplary user 53
Figure 8: Architecture of the *iManager* 54
Figure 9: Integration of the *iManager* in the user interface of the mobile device 55
Figure 10: Identity negotiation 56
Figure 11: Basic credential system protocols 58
Figure 12: De-anonymisation 60

11.2 Index of Tables

Table 1: Suggested structure of the database on IMS 18
Table 2: Class 1 IMS 22
Table 3: Class 2 IMS 23
Table 4: Class 3 IMS 24
Table 5: Description of requirements 39