WP42 - Universal Authentication Service

D42.1 - Requirements report

Document Identification	
Date	21.05.2013
Status	Final
Version	1.1

Related SP / WP	SP4/WP42	**Document Reference**	D42.1
Related Deliverable(s)	D21.2, D21.4, D32.1, D41.1	**Dissemination Level**	PU
Lead Participant	ULD	**Lead Author**	Meiko Jensen
Contributors	Moritz Horsch (TUD), Detlef Hühnlein (ECS)	**Reviewers**	Charles Bastos Rodríguez (ATOS) Sebastian Mödersheim (DTU)

This document is issued within the frame and for the purpose of the FutureID project. This project has received funding from the European Unions Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 318424

This document and its content are the property of the FutureID Consortium. All rights relevant to this document are determined by the applicable laws. Access to this document does not grant any right or license on the document or its contents. This document or its contents are not to be used or treated in any manner inconsistent with the rights or interests of the FutureID Consortium or the Partners detriment and are not to be disclosed externally without prior written consent from the FutureID Partners.

Each FutureID Partner may use this document in conformity with the FutureID Consortium Grant Agreement provisions.

Abstract

The present document provides a brief overview of existing authentication protocols and specifies requirements for the Universal Authentication Service (UAS) and the Authentication Protocol Specification (APS) language, which is to be interpreted by the UAS.

In order to provide an optimised message flow and enable the Claims Transformer Mode, the UAS will support all authentication protocols implemented by the various authentication tokens deployed across Europe. In comparison to the Dispatcher Mode, through which the Broker Service just forwards requests to existing Identity Providers, the Claims Transformer Mode enables session credentials and attribute-based credentials (ABC) issued by the UAS.

As the existing eID cards, eHealth cards and electronic signature cards already support a large variety of different authentication protocols and it may be expected that future authentication tokens will support other credentials and authentication protocols, it would be close to impossible to implement all required protocols using a conventional approach, because this would require a specialized program module for each and every authentication protocol.

In order to solve this problem, protocols are described by an appropriate Authentication Protocol Specification (APS) language which makes it possible to support arbitrary authentication protocols in a very efficient manner. The APS descriptions of the authentication protocols in turn refer to appropriate Basic Services, such as cryptographic primitives or smart card commands according to ISO/IEC 7816. As the different authentication protocols are all composed of a rather limited set of Basic Services, the problem of supporting arbitrary authentication protocols is reduced to providing this limited set of basic functionality and providing appropriate APS-descriptions for the different authentication protocols.

Document Information

History

Version	Date	Author	Changes
0.1	12.04.2013	Moritz Horsch	Created document structure
0.2	12.04.2013	Moritz Horsch	Added section 3.2
0.2	15.04.2013	Detlef Hühnlein	Added sections 2 and 3.1
0.3	15.04.2013	Moritz Horsch	Added abstract
0.4	29.04.2013	Meiko Jensen	Added more requirements (especially with respect to privacy)
0.5	06.05.2013	Detlef Hühnlein	Integrated input from ULD and created version for review.
1.0	21.05.2013	Detlef Hühnlein	Included review findings.
1.1	10.12.2013	Meiko Jensen	Fixed terminology changes.

Table of Contents

1. Introduction

1.1 Scope

The scope of this document is to provide the requirements for the Universal Authentication Service (UAS) and the formal Authentication Protocol Specification (APS) language.

1.2 Outline

The present document provides in Section 2 a brief overview of existing authentication protocols and specifies in Section 3 requirements for the Universal Authentication Service (UAS) and the Authentication Protocol Specification (APS).

1.3 Terminology

1.3.1 Key Words

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

1.3.2 Abbreviations and Notations

ABC	Attribute-based Credential
APS	Authentication Protocol Specification
BS	Basic Service
EAC	Extended Access Control
TLS	Transport Layer Security
UAS	Universal Authentication Service

2. Existing authentication procedures

The authentication of users, devices and services in combination with mechanisms for key agreement provides the foundation for secure communication and there are countless cryptographic protocols for this purpose (see [2], [3]). Even in an abstract perspective there are various standardized mechanisms for authentication (see [4]) and key management (see [5]). If one additionally considers the binding of the protocols to specific communication protocols[1] and the various possibilities to realize the personal security environment of a user[2] the situation becomes even more complex.

With respect to the eID sector the authentication protocols standardized in [6], [7], [8], [9] [10] are especially important, as they also include the specification of smart card commands according to [7] and it may be expected that these protocols turn out to be used in many eID-systems around the world, because they mechanisms are standardized by ISO and endorsed by ICAO for example.

[1] See http://www.ietf.org/rfc.html.
[2] Depending on the factors (knowledge, possession, biometrics, place [16], relationships [17] etc.) used for authentication there may be considerable differences.

3. Requirements

3.1 Universal Authentication Service

This Section describes the specific requirements for the Universal Authentication Service as outlined in Figure 1. Further requirements for the overall FutureID architecture, which in turn may induce additional requirements for the UAS, can be found in the deliverables of FutureID WP 22. Hence it should be noted that the list of requirements provided here may be seen as preliminary until the overall FutureID architecture and its corresponding requirements are final.

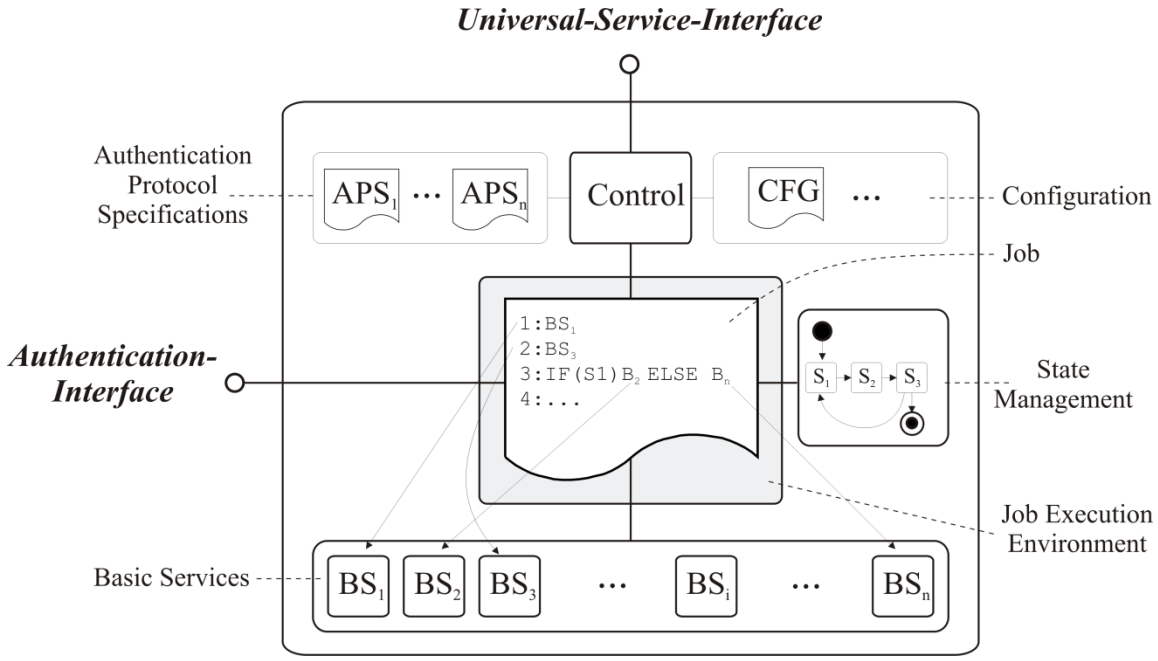

Figure 1: Outline of Universal Authentication Service

3.1.1 Interfaces

No.	IF-01 – Generic interfaces
Description	The UAS SHOULD provide generic interfaces, which support the transport of structured messages via a predefined set of message bindings.

No.	IF-02 – Authentication-Interface
Description	The UAS MUST provide an Authentication-Interface, which can be used by the FutureID-Client for example. The Authentication-Interface supports the PAOS-Binding

	and at least the following functions: • `StartPAOS` (see [11], [7]) • `DIDAuthenticate` (see [11], [7]) • `Transmit` (see [11], [7])

No.	IF-03 – Universal-Service-Interface
Description	The UAS MUST provide a Universal-Service-Interface, which can be used by the Broker Service for example. The Universal-Service-Interface supports the SOAP-Binding [12] and at least the following function: • `Authenticate` (see [8])

No.	IF-04 – Extensibility with respect to interfaces
Description	The UAS MUST provide a mechanism to add further interfaces at a later point in time.

No.	IF-05 – HSM-support
Description	The UAS MUST be able to use a hardware security module (HSM) for the secure storage and application of cryptographic keys.

3.1.2 Functional Requirements

No.	FU-01 – Execution of APS-specified authentication protocols
Description	The UAS MUST be able to execute arbitrary authentication protocols, which are described by an appropriate APS.

No.	FU-02 – Extensibility with respect to authentication protocols
Description	The UAS MUST provide a mechanism to add further authentication protocols at a later point in time.

No.	FU-03 – Extensibility with respect to basic services
Description	The UAS MUST provide a mechanism to add further basic services at a later point in time.

No.	FU-04 – APS development within FutureID
Description	The set of APS descriptions which is to be developed in FutureID SHOULD cover the authentication protocols supported by the existing eID cards identified in deliverable D32.1 [13]. This includes the following protocols: • EAC protocol [6] • RSA Authentication protocol according to [11] (part 7) • Mutual Authentication protocol according to [11] (part 7)

	• TLS protocol [14] In addition the following protocol MAY be supported: • CIPURSE v2 mutual authentication and key agreement protocol [15] •

No.	FU-05 – Error Handling
Description	The UAS SHOULD provide reasonable error handling, including sufficiently detailed error information in both machine-readable and human-readable format.

3.1.3 Privacy Requirements

No.	PR-01 – Support of privacy-specific requirements defined in APS
Description	The UAS MUST support the protocol- and privacy-specific requirements which are within the scope of the UAS and which may be defined within the APS.

No.	PR-02 – Protection of personally identifiable information
Description	The UAS or its operational environment MUST protect the processed personally identifiable information handled.

No.	PR-03 – No permanent storage of user attributes
Description	The UAS SHOULD not store user attributes in a permanent manner.

No.	PR-04 – Automated Logging of UAS events
Description	The UAS SHOULD create and maintain appropriate logs of all events of importance in the context of UAS transactions.

No.	PR-05 – Transparency and audit
Description	The UAS SHOULD provide an interface for forensics, audit and transparency requests, providing information on both its operations and processes and on all subsequent UAS events related to a certain (set of) transaction(s). It MUST be ensured that this interface is only available for authorized entities and that access to this interface is logged.

No.	PR-06 – Integrity of event logs
Description	If internal log files are used within the UAS, it MUST be ensured that the integrity of the event logs is protected in an adequate manner.

No.	PR-07 – No unauthorized access to event logs
Description	If internal log files are used within the UAS, it MUST be ensured that these event logs are only accessible by authorized entities.

No.	PR-08 – Redaction of event logs
Description	If internal log files are used within the UAS, these SHOULD provide means for redaction in case an individual claims its rights to deletion or correction of personal data.

3.2 Authentication Protocol Specification

This section describes the requirements for the Authentication Protocol Specification (APS) language.

3.2.1 Layered Approach

No.	APS_R1 – Layer Approach
Description	The APS language SHOULD use a layered approach to model the details of functions. At the first layer only a simple and abstract model of functions, e.g., send, encrypt, hash, should be used.

3.2.2 Notation

No.	APS_R2 – Simple and concise notation
Description	The APS language MUST provide a simple and concise notation.

3.2.2.1 Types

No.	APS_R3 – Constants and variables
Description	The APS language MUST support types for constants and variables.

3.2.2.2 Functions

No.	APS_R4 - Functions
Description	The APS language MUST support the definition of functions using different levels of abstraction ranging from an abstract specification (black box perspective) to concrete implementations, which allow execution of a function in some computational environment.

3.2.2.3 Messages

No.	APS_R5 – Encoding of messages
Description	The APS language MUST support types and encodings for constants, variables and messages such that a protocol implementation is able to fulfil a detailed technical specification.

No.	APS_R6 – Support of XML-encoding
Description	The APS language MUST support the XML-encoding of types and encodings for

	constants, variables and messages.

No.	APS_R7 – Support of other encodings
Description	The APS language SHOULD support other encodings for constants, variables and messages.

3.2.2.4 Channels

No.	APS_R8
Description	The APS language MUST support the modelling of channels for message exchange. It MUST be also possible to define different types of protection for channels (e.g. confidential and/or (unilaterally or mutually) authenticated channels) and bindings, which define how messages are transported via such a channel.

3.2.2.5 Goals

No.	APS_R9
Description	The APS language MUST support the modelling of goals that can be verified by model-checking tools.

3.2.3 Automatic Verification

No.	APS_R10
Description	The APS language MUST support the automatic verification with model-checking tools.

3.2.4 Cryptographic Primitives

No.	APS_R11 - Encryption
Description	The APS language MUST support the modelling of symmetric and asymmetric encryption.

No.	APS_R12 – Digital signatures and message authentication codes
Description	The APS language MUST support the modelling of the creation and verification of digital signatures and message authentication codes.

No.	APS_R13 – Hash functions
Description	The APS language MUST support the modelling of hash functions.

No.	APS_R14 – Extensibility with respect to cryptographic primitives
Description	The APS language MUST provide a mechanism to add further “user-defined” cryptographic primitives at a later point in time on an abstract and concrete level.

3.2.5 Privacy Properties

No.	APS-R15 – Privacy properties of protocols
Description	The APS language SHOULD be able to describe privacy properties of protocols as researched in WP 2.4.

4. Conclusion

The present document contains the specific requirements for the UAS and the APS language, which is to be interpreted by the UAS. Further requirements for the overall FutureID architecture, which in turn may induce additional requirements for the UAS, can be found in the deliverables of FutureID WP 22 (i.e. D22.1-D22.7).

As the overall FutureID system architecture (cf. Task 21.4) and the corresponding overall system requirements (cf. Tasks 22.1-22.7) are not finalized yet, the present list of requirements may be seen as preliminary until the overall FutureID architecture and its corresponding requirements are final. Nevertheless the requirements specified in the present document already provide a fairly solid foundation for the outline and design of the UAS and the APS language in the following tasks of WP 42.

5. Bibliography

[1] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels,* RFC 2119, 1997.

[2] A. J. Menezes, P. C. v. Oorschot and S. A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997.

[3] C. Boyd and A. Mathuria, Protocols for authentication and key establishment, 2003.

[4] ISO/IEC, *Information Technology - Security Techniques - Entity Authentication,* ISO/IEC 9798, Part 1 - 5.

[5] ISO/IEC, *Information Technology - Security Techniques - Key Management,* ISO/IEC 11770, Parts 1 - 4, 1996 - 2008.

[6] Bundesamt für Sicherheit in der Informationstechnik (BSI), *Advanced Security Mechanisms for Machine Readable Travel Documents,* Technical Guideline TR-03110, Version 2.10, Part 1 - 3, https://www.bsi.bund.de/ContentBSI/EN/Publications/Techguidelines/TR03110/BSITR03110.html, 2012.

[7] ISO/IEC, *Identification cards - Integrated circuit card programming interfaces,* ISO/IEC 24727, Part 1 - 5.

[8] Comité Européen de Normalisation (CEN), Identification card systems - European Citizen Card, Part 1 - 4, CEN/TS 15480, 2008.

[9] Comité Européen de Normalisation (CEN), *Application Interface for smart cards used as Secure Signature Creation Devices - Part 1: Basic services,* CEN/TS 14890-1, 2008.

[10] Comité Européen de Normalisation (CEN), *Application Interface for smart cards used as Secure Signature Creation Devices - Part 2: Additional Services,* CEN/TS 14890-2, 2008.

[11] Bundesamt für Sicherheit in der Informationstechnik (BSI), *eCard-API-Framework,* Technical Guideline TR-03112, Part 1 - 7, Version 1.1.2, https://www.bsi.bund.de/ContentBSI/Publikationen/TechnischeRichtlinien/tr03112/index_htm.html.

[12] B. Don, E. David, K. Gopal, L. Andrew and M. Noah, Simple Object Access Protocol

(SOAP) 1.1, 2000.

[13] FutureID, *Survey and analysis of existing eID and credential systems,* Deliverable 32.1, 2013.

[14] T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.2,* 2008.

[15] OSPT Alliance, *CIPURSE V2,* Revision 1.0, 28.09.2012, Cryptographic Protocol.

[16] E. Bertino and M. S. Kirkpatrick, “Location-Aware Authentication and Access Control,” in *AINA*, 2009.

[17] J. G. Brainard, A. Juels, R. L. Rivest, M. Szydlo and M. Yung, “Fourth-factor authentication: somebody you know,” in *ACM Conference on Computer and Communications Security*, 2006.