

WP-02-03

PL EUDI Wallet Certification System

Security Requirements for Wallet Secure
Cryptographic Application for Digital Identity
Wallets

VERSION 1.01

2026.07.23

Document reference	WP-02-03
Version	1.01
Status	FINAL
Date	2026.07.24
Document type	Requirements
Parent framework	GP-02
Scheme Owner	Republic of Poland / Minister of Digital Affairs
Certification target	WSCA
Primary audience	EUDI Wallet Provider
Secondary audience	CAB-ITSEF, CAB-CB
Assurance level <evaluation>	EAL 4 augmented by AVA_VAN.5
Subsidiary Documents	WM-02-01, WM-02-02

Contents

Page

Foreword.....	4
Normative references.....	5
Terms and definitions	6
1 Introduction	7
1.1 Document Reference	7
1.2 Document Overview	7
1.2.1 TOE Type.....	9
1.2.2 TOE functionality	10
2 Security problem definition.....	11
2.1 Assets	11
2.1.1 Primary assets	11
2.1.2 TSF data.....	11
2.2 Subjects	12
2.3 Threats	12
2.4 Organizational security policies	13
2.5 Assumptions	13
3 Security objectives	15
3.1 Security objectives for the TOE.....	15
3.2 Security objectives for the operational environment	16
3.3 Security objectives rationale	16
4 Security requirements	19
4.1 Typographic conventions	19
4.2 Security functional requirements.....	19
4.2.1 Security Audit (FAU).....	19
4.2.2 Cryptographic Support (FCS).....	20
4.2.3 User data protection (FDP)	22
4.2.4 Identification and authentication (FIA)	27
4.2.5 Security Management (FMT)	33
4.2.6 Protection of the TSF (FPT).....	36
4.2.7 Trusted path/channels (FTP).....	37
4.3 Security assurance requirements.....	38
4.4 Security requirements rationale.....	39
4.4.1 Coverage.....	39
4.4.2 Rationale.....	41
4.4.3 SFR Dependencies	42
Bibliography	45

Foreword

This document specifies a security requirement for Wallet Secure Cryptographic Application (WSCA) component in Polish certification scheme for EU Digital Identity Wallet.

It builds upon the draft of Protection profile for Wallet Secure Cryptographic Application for digital Identity Wallets by Technical Committee CEN/TC 224 "Personal identification, electronic signature and cards and their related systems and operations", WG17 "Protection profiles in the context of SS CD".

Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 15408-1:2022 Information security, cybersecurity and privacy protection - Evaluation criteria for IT security - Part 1: Introduction and general model.

ISO/IEC 15408-2:2022 Information security, cybersecurity and privacy protection - Evaluation criteria for IT security - Part 2: Security functional components.

ISO/IEC 15408-3:2022 Information security, cybersecurity and privacy protection - Evaluation criteria for IT security - Part 3: Security assurance components.

ISO/IEC 15408-5:2022 Information security, cybersecurity and privacy protection - Evaluation criteria for IT security - Part 5: Pre-defined packages of security requirements.

NOTE The following are equivalent to the ISO/IEC 15408 standards:

Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, CC:2022, Revision 1, CCMB-2022-11-001, November 2022.

Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, CC:2022, Revision 1, CCMB-2022-11-002, November 2022.

Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, CC:2022, Revision 1, CCMB-2022-11-003, November 2022.

Common Criteria for Information Technology Security Evaluation, Part 5: Pre-defined packages of security requirements, CC:2022, Revision 1, CCMB-2022-11-003, November 2022.

Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 15408-1, and [IA5a], and the following apply.

NOTE In the context of this document, the following terms:

- Target of Evaluation (TOE) defined in [CC1] and
- Wallet Secure Cryptographic Application (WSCA) defined in [IA5a]

are synonyms.

Definitions:

WalletProviderAuthenticationData

Data used by the WSCA to authenticate the Wallet Provider.

WalletInstanceAuthenticationData

Data used by the WSCA to authenticate the Wallet Instance.

UserAuthenticationData

Data used by the WSCA to authenticate the Wallet User.

WSCDAuthenticationData

Data used by the WSCA to authenticate to WSCD and get access to its resources.

Acronyms:

PID	Personal Identification Data
TOE	Target of Evaluation
TSF	TOE Security Functionality

1 Introduction

1.1 Document Reference

Title	Security Requirements for Wallet Secure Cryptographic Application
CC	CC:2022, Revision 1
Version	1.0
Date	2026.06.12
Keywords	EUDIW, WSCA, security requirements

1.2 Document Overview

The EU Digital Identity Wallet (EUDIW) uses Wallet Secure Cryptographic Devices (WSCD) as tamper-resistant devices to protect critical assets and securely execute cryptographic functions.

The Wallet Secure Cryptographic Application (WSCA) was introduced in [IA5c] as an application that manages critical assets by being linked to and using cryptographic and non-cryptographic functions provided by the WSCD. The WSCA exposes a Secure Cryptographic Interface (SCI) to the Wallet Instance.

Critical assets are defined in [IA5c] and mean assets within or in relation to a wallet unit of such extraordinary importance that where their availability, confidentiality or integrity are compromised, this would have a very serious, debilitating effect on ability to rely on the wallet unit. Critical assets encompass data related to the management of Personal Identification Data (PID) and consist of secret key material and TOE Security Functionality (TSF) data with authentication secrets and data controlling cryptographic operations. Other data or keys, besides those used in connection with PID may also be considered as critical assets.

The WSCA is a Target of Evaluation (TOE) for this document.

As stated in [eIDAS], Article 5a(5), the EUDIW shall provide an electronic identification means meeting the requirements set out in [eIDAS], Article 8 with regards to assurance level high. The functionality of WSCA is further defined in [IA5a] and includes:

- (a) perform wallet cryptographic operations involving critical assets other than those needed for the wallet unit to authenticate the wallet user only in cases where those applications have successfully authenticated wallet users;
- (b) where they authenticate wallet users in the context of performing electronic identification at assurance level high; perform authentication of wallet users, in accordance with, the

requirements for the characteristics and design of electronic identification means at assurance level high, as set out in [CIR 2015/1502];

- (c) are able to securely generate new cryptographic keys;
- (d) are able to perform secure erasure of critical assets;
- (e) are able to generate a proof of possession of private keys;
- (f) protect the secret keys generated by those wallet secure cryptographic applications during the existence of the keys;
- (g) comply with the requirements for the characteristics and design of electronic identification means at assurance level high, as set out in [CIR 2015/1502];
- (h) are the only components able to execute wallet cryptographic operations and any other operation with critical assets in the context of performing electronic identification at assurance level high.

The scope of the current document is limited to such functionality, while an ST may extend the functionality of the TOE.

The document defines a baseline set of requirements. However, the ST author may refine these requirements and/or introduce requirements specific to the TOE and/or TOE operational environment. Any refined and added requirement must be reflected consistently and must not reduce or contradict the mandated security requirements for the TOE (unless explicitly allowed via application note).

The WSCA provides EUDIW with functionality to support management and use of critical assets. As part of granting a user access to critical assets and PID keys in particular, the user is authenticated by the TOE. This authentication is aimed at supporting the EUDIW provider to provide an electronic identification means.

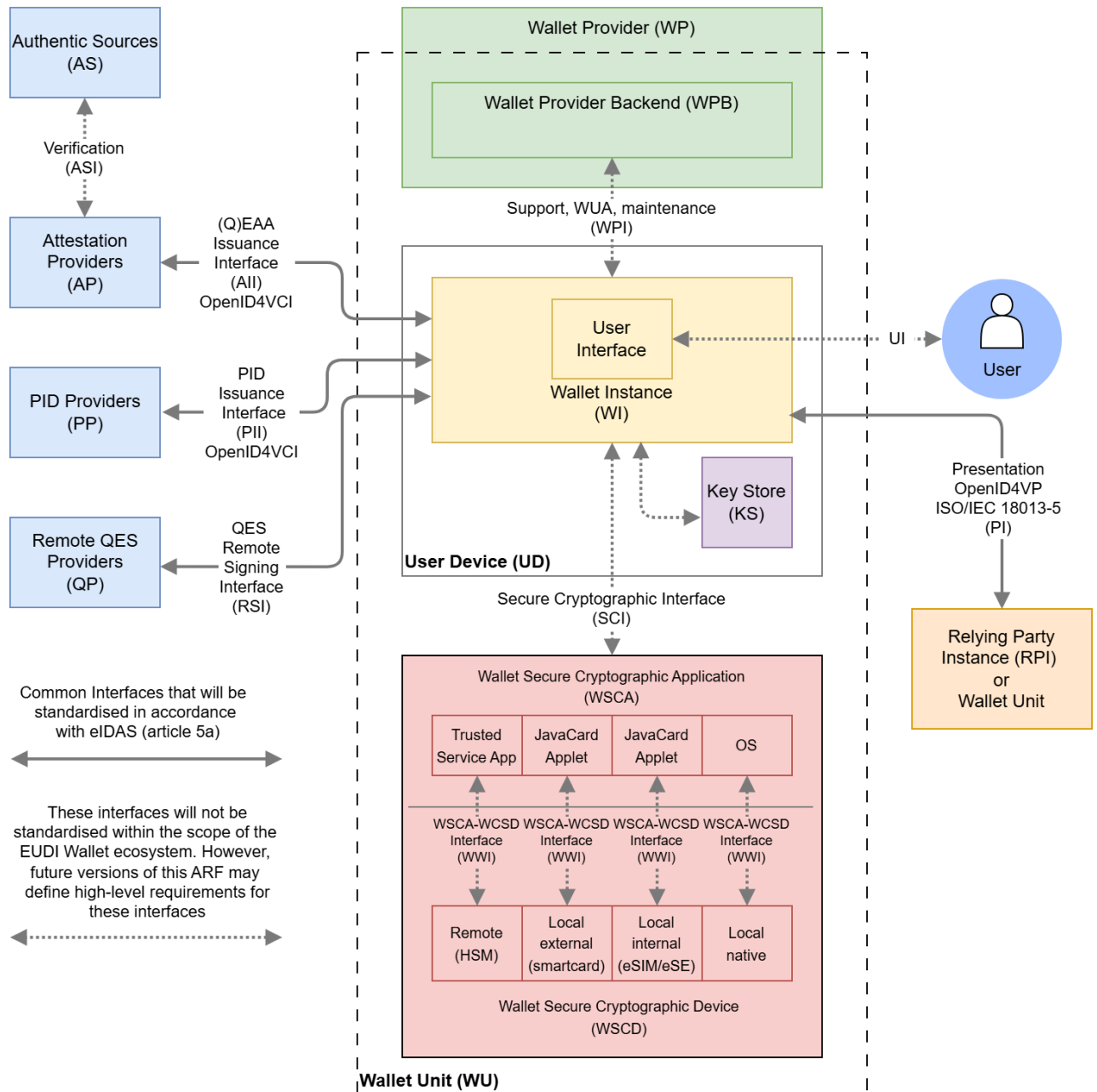


Figure 1 EUDIW ecosystem reference architecture [ARF]

1.2.1 TOE Type

As mentioned above, the TOE is an application that manages critical assets using the capabilities of the WSCD.

The TOE and the WSCD can be deployed in several different ways. This document assumes a fully remote WSCD deployment, where the Wallet Instance running on user devices accesses the WSCA over the Internet to manage critical assets.

1.2.2 TOE functionality

The WSCA provides the following functions to be used through the Wallet Instance:

- WSCA serves multiple end-users. There is a function (`create_new_user1`) defined to establish a user association within WSCA with user authentication data and bind it to the specific Wallet Unit. The function shall be used by a Wallet Instance.
- WSCA supports single Wallet Provider. The WSCA ensures authentication is successfully performed prior to allowing any functions associated with the wallet user, Wallet Provider, or Wallet Instance to be executed:
 - The WSCA shall authenticate users. Authentication data is provided to the TOE through the Wallet Instance. The user authentication data is to be based on at least two factors. One authentication factor must be verified by the TOE. Other authentication factors may be verified outside of TOE, in which case the TOE shall ensure that it receives and verifies evidence from the external party that verification has taken place.
 - The WSCA shall authenticate Wallet Provider such that only the legitimate Wallet Provider gains access to the WSCA.
 - The WSCA shall authenticate and communicate securely with Wallet Instances such that only legitimate Wallet Instances gain access to the WSCA.
- WSCA shall authenticate and communicate securely with the WSCD such that only legitimate WSCDs are used by the WSCA.
- WSCA shall provide the function for the Wallet Provider to manage any necessary TSF data.
- WSCA shall manage cryptographic keys associated with a user. The keys are generated and protected by WSCD. It provides functions (`create_new_key1`, `delete_key1`) for the Wallet Instance to create and delete user keys on the WSCD.
- The keys associated with a user shall be used for cryptographic operations. There is a function (`cryptographic_operations1`) for cryptographic operations with user keys.

The WSCA does not have a user interface, and any user authentication credentials are to be provided through the Wallet Instance. The ability of the WSCA to authenticate users may rely on information received from other components or services. This includes local Wallet Instance components which provide evidence for e.g. biometric verification or remote Wallet Provider electronic authentication services.

This document describes security requirements for a Wallet Secure Cryptographic Application. The actual interface to use the functions of the WSCA is not included and left for manufacturers to design and implement.

¹ The function name indicates the operation defined in (iteration of) FDP_ACC.1 security function requirement (see section 4.2.3 for details).

2 Security problem definition

2.1 Assets

2.1.1 Primary assets

R.SecretKey: private keys used in asymmetric or symmetric cryptographic functions, managed and used by the TOE in support of the cryptographic services that it offers. This includes user keys, owned and used by specific users used in the implementation and operation of the TOE. The TOE relies on a WSCD for actual key protection and use of the key within the WSCD. The integrity of these keys shall be protected.

Application Note 1

Depending on the implementation, the R.SecretKey may be, for example, an identifier of the secret key inside the WSCD or an encrypted key blob (stored externally to WSCD). In both cases the plain key value of R.SecretKey is protected in integrity and confidentiality by the WSCD.

Application Note 2

The integrity of the cryptographic key material is ensured by WSCD. The TOE shall ensure the integrity of key identifiers, encrypted key blobs, user binding as well as associated metadata.

R.PublicKey: public keys managed and used by the TOE in support of the cryptographic services that it offers. The integrity of these keys shall be protected.

One user is expected to have multiple keys or key pairs managed by the WSCA.

R.OperationsData: set of data which is transmitted by S.WalletInstance to the TOE for digital operation carried out on behalf of the user including the data that is the result of the digital operation. The R.OperationsData shall be protected in integrity

Application Note 3

The document defines a baseline set of assets requiring protection. However, the ST author may introduce additional assets specific to the TOE. Any added asset shall be reflected consistently in the Security Problem Definition, corresponding security objectives, and the selected or refined SFRs. Additional assets may not reduce or contradict the mandated security requirements, and the ST shall provide clear rationale showing how these assets are protected.

2.1.2 TSF data

In addition to the assets described above, the TOE also has TSF data used for authentication in different scenarios. The TSF data is:

- R.WalletProviderAuthenticationData is used by the TOE to authenticate the S.WalletProvider.
- R.WalletInstanceAuthenticationData is used by the TOE to authenticate the S.WalletInstance.
- R.UserAuthenticationData is used by the TOE to authenticate the S.WalletUser.

- R.WSCDAuthenticationData is used by the TOE to access the WSCD.

Application Note 4

The R.WSCDAuthenticationData shall include elements for mutual identification and authentication between WSCA and WSCD. This data is denoted as R.WSCDAuthenticationData-WalletProviderElement for direct reference.

The R.WSCDAuthenticationData shall also include data elements which are unique for a user or specific key. The WSCA uses a cryptographic module being WSCD, which has been certified to meet the requirements in [CEN EN PP 419 221-5]. The R.WSCDAuthenticationData shall include Authorisation Data as defined in [CEN EN PP 419 221-5]. This data is denoted as R.WSCDAuthenticationData-WalletUserElement for direct reference.

2.2 Subjects

S.WalletProvider: a Wallet Provider backed application that communicates with the TOE (directly or through the Wallet Instance) and/or local Wallet Provider Administrator. This subject manages necessary TSF data (including authentication data and/or configuration data if applicable) and may request operations on the TOE.

Application Note 5

The ST may distinguish two subjects, related to Wallet Provider backed application and related to Wallet Provider Administrator. This must be reflected consistently and must not reduce or contradict the mandated security requirements for the TOE (unless explicitly allowed via application note).

S.WalletInstance: a Wallet application that communicates with the TOE over a external network interface. This subject requests operations on the TOE.

S.WalletUser: an end user of the TOE which is required to be authenticated before any cryptographic operation can be carried out with the users' keys. The S.WalletUser does not manage or use keys directly as such operations are mediated by the S.WalletInstance. An end user communicates with the TOE by using a Wallet application (S.WalletInstance).

The TOE has a number of subjects of the type S.WalletUser (the TOE is assumed to be deployed remotely to S.WalletInstance).

2.3 Threats

The following threats are defined for the TOE.

T.Copy

An attacker copies R.SecretKey(s) outside the TOE and uses them without the user's consent.

T.Derive

An attacker derives the plain value of the key referenced by R.SecretKey from the Public Key and uses them without the user's consent.

Application Note 6

This threat relates to whether the WSCA instructs the WSCD to use a weak key generation mechanism. Guidance on cryptographic algorithms can be found in [EUCC ACM].

T.Modify

An attacker modifies TSF authentication data and circumvents the security functions of the WSCA.

T.Misuse

An attacker gains access to an operation function (cryptographic_operations) of the TOE and uses it without the user's consent.

T.Forge_WalletInstance_communication

An attacker can read, change, forge or replay communication between the Wallet Instance and the TOE. Sensitive data may be lost or modified, or Cryptographic Operations may be carried out on data other than the data intended by the S.WalletUser.

T.Forge_WSCD_communication

An attacker can read, change, forge or replay communication between TOE and the WSCD. Sensitive data may be lost or modified, or operations on the WSCD may be carried out on data other than the data intended by the S.WalletUser.

2.4 Organizational security policies

OSP.CRYPTO

The TOE shall only use algorithms, algorithm parameters and key lengths endorsed by recognized authorities. This includes algorithms invoked by TOE on WSCD as well as algorithms used by TOE for the integrity and confidentiality of TOE assets.

Application Note 7

The algorithms listed [EUCC ACM] serve as appropriate candidates.

2.5 Assumptions

A.Certified_Wallet

The Wallet Instance is assumed to be certified or being under certification process according to [IA5c]. The Wallet Instance is assumed to enforce cybersecurity precautions to ensure that authentication credentials and user data for cryptographic operations provided through the Wallet Instance to the TOE

are protected. This also means that the communication end point from the Wallet Instance to the TOE is protected (including TOE authentication by Wallet Instance).

A.Conformant_WSCD

The WSCD is assumed to be conformant to the applicable requirements in [IA5c] to secure cryptographic key generation, deletion, protection and operation. The WSCD shall provide capabilities for communication using secure channels with the TOE. The WSCD is assumed to be certified to meet the requirements of [CEN EN PP 419 221-5], and configured, and used accordingly.

A.WalletUser_Protection

S.WalletUser is assumed to protect their authentication credentials against use by others. S.WalletProvider is assumed to restrict admissible R.UserAuthenticationData such that S. WalletUser can reliably protect their user authentication credentials.

A.WalletProvider_Protection

S.WalletProvider is assumed to protect their user authentication credentials against use by others.

Application Note 8

The ST shall extend the A.WalletProvider_Protection assumption according to Wallet Provider roles and functions on the TOE. The ST shall refine the SPD accordingly. Any added assumption must be reflected consistently and must not reduce or contradict the mandated security requirements for the TOE (unless explicitly allowed via application note).

A.Secure_Environment

It is assumed that the TOE is operated in a secure physical and operational environment, protected against unauthorized access, tampering, and environmental threats, and it communicates with a genuine Wallet Instance and WSCD, if authenticated.

A.External_Authentication

It is assumed that if part of the user authentication is carried outside the TOE, the external verification meets the applicable requirements from [CIR 2015/1502].

3 Security objectives

3.1 Security objectives for the TOE

OT.Data_Protection

The TOE shall protect its assets and data such that their values cannot be copied or for secret data inferred outside the TOE, cannot be modified or injected by an attacker.

OT.Algorithms

The TOE shall rely on a WSCD for R.SecretKey generation, usage and deletion. All algorithms (including algorithm parameters and key lengths) used by TOE for TSF protection and selected by the TOE and used by the WSCD shall be endorsed by recognized authorities and such that public information cannot be used to derive private keys or compromise integrity or confidentiality of data.

Application Note 8

The TOE shall rely on WSCD for all handling of plain R.SecretKey (generation, usage and deletion) and its protection if stored externally to WSCD. See also Application Note 1 and Application Note 2.

OT.Authentication

The TOE shall carry out an authentication check on all subjects before allowing them to use the TOE. In particular, the TOE shall always require authentication before creating S.WalletUser object, and creating, using (cryptographic_operations), and deleting a S.WalletUser secret key.

OT.Communication_Protection

The TOE shall provide communication over secure channels (mutual authentication and channel binding) between the Wallet Instance and WSCD to protect the confidentiality and integrity of sensitive data during transmission. This includes protection against replay of transmission data.

The table below maps how the WSCA supports the Wallet solution meeting the objectives describing in section 1.2.

Security function	Related WSCA security objectives.
(a) perform wallet cryptographic operations involving critical assets other than those needed for the wallet unit to authenticate the wallet user only in cases where those applications have successfully authenticated wallet users;	OT.Algorithms and OT.Authentication
(b) where they authenticate wallet users in the context of performing electronic identification at assurance level high; perform authentication of wallet users, in accordance with, the requirements for the characteristics and design of electronic identification means at assurance level high, as set out in Implementing Regulation (EU) 2015/1502;	OT.Authentication

(c) are able to securely generate new cryptographic keys;	OT.Algorithms
(d) are able to perform secure erasure of critical assets;	OT.Algorithms
(e) are able to generate proof of possession of private keys;	OT.Algorithms
(f) protect the secret keys generated by those wallet secure cryptographic applications during the existence of the keys;	OT.DataProtection, OT.Communication_Protection
(g) comply with the requirements for the characteristics and design of electronic identification means at assurance level high, as set out in Implementing Regulation (EU) 2015/1502;	All OT. Note the role of the WSCA is not to perform Wallet User identification.
h) are the only components able to execute wallet cryptographic operations and any other operation with critical assets in the context of performing electronic identification at an assurance level high	OT.Authentication and OT.Algorithms

3.2 Security objectives for the operational environment

OE.WSCD

The WSCD shall provide cryptographic key generation, key protection, cryptographic operation and cryptographic key destruction. The WSCD is expected to provide communication over secure channels with the TOE with mutual authentication. The WSCD shall be certified to meet the requirements of [CEN EN PP 419 221-5] and configured and used accordingly.

OE.Environment

TOE shall be operated in a secure physical and operational environment, protected against unauthorized access, tampering, and environmental threats. The authenticated Wallet Provider is trusted and his authentication data is confidential.

OE.Wallet

The Wallet Instance shall be certified according to [GP-02]. The authentication credentials and user data provided through the Wallet Instance to the TOE are trusted (protected by Wallet Instance and Wallet User). The user authentication data is restricted by Wallet Provider to secure admissible values, the data verified outside the TOE is trusted and meets the applicable requirements from [CIR 2015/1502]. The Wallet Instance communication endpoint to the TOE is protected and supports mutual authentication.

3.3 Security objectives rationale

The table below shows the mapping of Threats, Organisational Security Policies and Assumptions to Security Objectives for the TOE and for the TOE Environment.

	OT.Data_Protection	OT.Algorithms	OT.Authentication	OT.Communication_Protection	OE.WSCD	OE.Environment	OE.Wallet
T.Copy	X	X			X		
T.Derive		X			X		
T.Modify	X					X	
T.Misuse			X				
T.Forge_WalletInstance_communication				X			X
T.Forge_WSCD_communication				X	X		
OSP.Crypto		X			X		
A.Certified_Wallet							X
A.Conformant_WSCD					X		
A.WalletUser_Protection							X
A.WalletProvider_Protection						X	
A.Secure_Environment					X	X	X
A.External_Authentication							X

Table 1 Security objectives mapping

T.Copy is addressed by the OT.Algorithms requiring that the TOE shall rely on a WSCD for R.SecretKey protection covering generation, usage and deletion and by OE.WSCD requiring that WSCD provides appropriate secure algorithms. It is also addressed by OT.Data_Protection requiring that TOE keeps assets and data protected .

T.Derive is addressed by the OT.Algorithms requiring that all algorithms (including algorithm parameters and key lengths) selected by the TOE and used by the WSCD shall be endorsed by recognized authorities and such that public information cannot be used to derive private keys or compromise integrity or confidentiality of data. It is also addressed by OE.WSCD requiring that WSCD is certified and provides appropriate secure algorithms.

T.Modify is addressed by OT.Data_Protection requiring TOE to keep assets and data protected. Further on, it is addressed OE.Environment requiring that TOE physical environment is secure and protected against unauthorized access, tampering, and environmental threats.

T.Misuse is addressed by the requirement in OT.Authentication that all subjects are authenticated before they can use the TOE.

T.Forge_WalletInstance_communication is addressed by OT.Communication_Protection requiring secure channel with mutual authentication and replay protection between TOE and Wallet Instance This is supported by OE.Wallet requiring that the Wallet Instance communication endpoint to the TOE is protected and supports mutual authentication.

T.Forge_WSCD_communication is addressed by OT.Communication_Protection requiring secure channel with mutual authentication and replay protection between TOE and Wallet Instance. This is supported by OE.WSCD requiring that the WSCD provides secure channel from the TOE with mutual authentication.

OSP.Crypto is addressed by the OT.Algorithms requiring that all algorithms (including algorithm parameters and key lengths) used by TOE for TSF protection and selected by the TOE and used by the WSCD shall be endorsed by recognized authorities. It is also addressed by OE.WSCD requiring that WSCD is certified and provides appropriate secure algorithms.

A.Certified_Wallet is addressed by OE.Wallet requiring that: Wallet Instance is certified; authentication credentials and user data provided through the Wallet Instance to the TOE are trusted (protected by Wallet Instance and Wallet User); the Wallet Instance communication endpoint is protected and supports mutual authentication.

A.Conformant_WSCD is upheld by OE.WSCD.

A.WalletUser_Protection is addressed by OE.Wallet requiring that: the authentication credentials and user data provided through the Wallet Instance to the TOE are trusted (protected by Wallet Instance and Wallet User); and the user authentication data is restricted by Wallet Provider to secure admissible values.

A.WalletProvider_Protection is addressed by OE.Environment requiring that the authenticated Wallet Provider is trusted and his authentication data is confidential

A.Secure_Environment is addressed by OE.Environment requiring that TOE is operated in a secure physical and operational environment, protected against unauthorized access, tampering, and environmental threats. It is also addressed by OE.Wallet and OE.WSCD requiring that Wallet Instance and WSCA are certified and provide mutual authentication for communication with TOE

A.External_Authentication is addressed by OE.Wallet requiring that the user authentication verified outside the TOE is trusted and meets the applicable requirements from [CIR 2015/1502].

4 Security requirements

4.1 Typographic conventions

The following conventions are used in the definitions of the SFRs:

Refinements made in this document are always updates of the text for the SFR. They are marked in bold and the original text is indicated as strikeouts.

Selections made in this document are written in bold italics, and the original text is indicated in a footnote.

Selections that are left to be filled in by the Security Target author appear in square brackets with an indication that a selection is to be made, [selection:], and the description of selections options are written in normal font.

Assignments made in this document are written in italics, and the original text is indicated in a footnote. Assignments that are left to be filled in by the Security Target author appear in square brackets with an indication that an assignment is to be made, [assignment:], and the assignment description is written in normal font.

Iterations are denoted by a slash "/" and the iteration indicator after the component identifier.

The Security Target shall mark all operations directly following [CC2].

4.2 Security functional requirements

4.2.1 Security Audit (FAU)

FAU_GEN.1 Audit data generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shutdown of the audit functions;
- b) All auditable events for the [selection, choose one of: minimum, basic, detailed, not specified] level of audit; and
- c) [assignment: other specifically defined auditable events].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

- a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
- b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [assignment: other audit relevant information].

Application Note 9

The auditable events shall cover at least the following:

- *authentication failures related to (iterations of) FIA_AFL.1;*

- operations defined by (iterations of) FDP_ACF.1/FDP_ACC.1;
- changing TSF data by S.WalletProvider defined by FMT_MTD.1/WalletProvider.

FAU_GEN.2 User identity association

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

4.2.2 Cryptographic Support (FCS)

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: cryptographic key generation algorithm] and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

Application Note 10

The ST shall include an iteration of this SFR for every key type for:

- keys generated on WSCD on TOE request and associated with FDP_ACC.1/KeyCreation and, if applicable, with FDP_ACC.1/UserCreation
- other support keys generated by TOE itself for TSF data protection.

The ST shall cover the dependencies of each iteration of this SFR accordingly (TOE implementation dependent).

Application Note 11

In case of the keys generated on WSCD on TOE request, the TOE is expected to use a cryptographic module certified in conformance with [EN 419 221-5], see also OE.WSCD for key generation. Although TSF does not generate keys itself, this SFR expresses the requirement for the TSF to invoke the cryptographic module with the appropriate parameters whenever key generation is required.

Application Note 12

In case of the keys generated by TOE itself for TSF data protection, the ST TSS shall briefly describe the general key aim. This description may be high-level and concise (e.g. protection of R.WSCDAuthenticationData), the specifics shall be described in ADV_TDS evidence.

FCS_CKM.6 Timing and event of cryptographic key destruction

- FCS_CKM.6.1 The TSF shall destroy [assignment: list of cryptographic keys (including keying material)] when [selection: no longer needed, [assignment: other circumstances for key or keying material destruction]].
- FCS_CKM.6.2 The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method [assignment: cryptographic key destruction method] that meets the following: [assignment: list of standards].

Application Note 13

The ST shall cover destruction of keys generated on WSCD on TOE request (associated with FDP_ACC.1/KeyDeletion) and other support keys generated (or imported) by TOE itself applying iteration of this SFR accordingly.

The ST shall cover the dependencies of each iteration of this SFR accordingly (TOE implementation dependent).

Application Note 14

In case of the keys generated on WSCD on TOE request, the TOE is expected to use a cryptographic module certified in conformance with [EN 419 221-5] for key destruction.

Although the TSF may not destruct keys, this SFR expresses the requirement for the TSF to invoke the cryptographic module with the appropriate parameters whenever key destruction is required.

The Security Target must specify the method(s) of secure destruction of all secret keys and all support keys, and shall ensure that all are covered by a secure destruction method. If necessary, then more than one iteration of FCS_CKM.6 may be included to describe different standards for secure deletion. The 'list of standards' in the final assignment may be met in the Security Target by simply providing a description of the action taken to zeroise the keys rather than referencing an external standard.

FCS_COP.1 Cryptographic operation

- FCS_COP.1.1 The TSF shall perform [assignment: list of cryptographic operations] in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithm] and cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

Application Note 15

The ST shall cover cryptographic operations performed on WSCD on TOE request (associated with FDP_ACC.1/CryptoOperations) and other support keys generated/imported/derived by TOE itself applying iteration of this SFR accordingly.

The ST shall cover the dependencies of each iteration of this SFR accordingly (TOE implementation dependent).

Application Note 16

In case of the keys generated on WSCD on TOE request, the TOE shall use a cryptographic module certified in conformance with [EN 419 221-5] for cryptographic operations.

4.2.3 User data protection (FDP)

FDP_ACC.1/UserCreation	Subset access control
------------------------	-----------------------

FDP_ACC.1.1/UserCreation The TSF shall enforce the *UserCreation SFP*² on:

Subjects: S.WalletInstance

Objects: S.WalletUser

Operations: create_new_user.

*The TOE creates S.WalletUser with information provided by the subject*³

FDP_ACF.1/UserCreation	Security attribute based access control
------------------------	---

FDP_ACF.1.1/UserCreation

The TSF shall enforce the *UserCreation SFP*⁴ to objects based on the following:

*(1) whether the subject is a S.WalletInstance authorised to create a new S.WalletUser*⁵.

FDP_ACF.1.2/UserCreation

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

(1) only an authenticated S.WalletInstance who has been authorised for creation of new users can carry out the create_new_user operation

*(2) [assignment: other rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]*⁶.

² [assignment: access control SFP]

³ [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]

⁴ [assignment: access control SFP]

⁵ [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]

⁶ [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

FDP_ACF.1.3/UserCreation

The TSF shall explicitly authorise access of subjects to objects based on the following additional rules:

(1) [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*].

FDP_ACF.1.4/UserCreation

The TSF shall explicitly deny access of subjects to objects based on the following additional rule: [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*].

Application Note 17

The ST TSS shall briefly describe how the TOE verifies, based on SFP-relevant security attributes, that S.WalletInstance is authorised to create a new S.WalletUser.

Application Note 18

In the FDP_ACF.1.4, the ST shall explicitly define any applicable rules that deny access of subjects to objects related to actions defined in (iterations of) FIA_AFL.1. E.g. if FIA_AFL.1 defines blocking of a subject, then ST shall state if blocked subject is denied access to objects covert by this access control SFP

FDP_ACC.1/KeyCreation Subset access control
--

FDP_ACC.1.1/KeyCreation The TSF shall enforce the KeyCreation SFP⁷ on:

Subjects: S.WalletInstance and S.WalletUser.

Objects: The assets R.SecretKey and for asymmetric cryptography, R.PublicKey assigned to S.WalletUser.

Operations: create_new_key.

The S.WalletInstance of S.WalletUser instructs the TOE to instruct the WSCD to generate a new key or key pair. After the key or key pair is generated, the TOE assigns it to S.WalletUser⁸.

Application Note 19

The ST shall specify the different types of cryptographic keys linked to S.WalletUser, which can be generated by WSCD on TOE request with (iterations of) FCS_CKM.1.

Application Note 20

The ST TSS shall briefly describe the usage context of cryptographic keys linked to S.WalletUser, which can be generated by WSCD on TOE request.

⁷ [assignment: *access control SFP*]

⁸ [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

- FDP_ACF.1.1/KeyCreation The TSF shall enforce the *KeyCreation SFP*⁹ to objects based on the following:
- (1) *whether the subject is a S.WalletInstance authorised to create a new key or key pair linked to S.WalletUser*
 - (2) *[assignment: other rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]*¹⁰.
- FDP_ACF.1.2/KeyCreation The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:
- (1) *only an authenticated S.WalletInstance who has been authorised for creation of new key or key pairs, linked to S.WalletUser, can carry out the Create_Key operation*¹¹.
- FDP_ACF.1.3/KeyCreation The TSF shall explicitly authorise access of subjects to objects based on the following additional rules:
- [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects].*
- FDP_ACF.1.4/KeyCreation The TSF shall explicitly deny access of subjects to objects based on the following additional rule:
- (1) *[assignment: rules, based on security attributes, that explicitly deny access of subjects to objects].*

Application Note 21

The ST TSS shall briefly describe how the TOE verifies, based on SFP-relevant security attributes, that a S.WalletInstance is authorised for creation of new key or key pairs linked to S.WalletUser.

Application Note 22

In the FDP_ACF.1.4, the ST shall explicitly define any applicable rules that deny access of subjects to objects related to actions defined in (iterations of) FIA_AFL.1. E.g. if FIA_AFL.1 defines blocking of a subject, then ST shall state if blocked subject is denied access to objects covert by this access control SFP

⁹ [assignment: access control SFP]

¹⁰ [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]

¹¹ [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

FDP_ACC.1.1/KeyDeletion The TSF shall enforce the *KeyDeletion SFP*¹² on:

Subjects: S.WalletInstance and S.WalletUser.

Objects: The assets R.SecretKey and for asymmetric cryptography, R.PublicKey linked to S.WalletUser.

Operations: delete_key.

*The S.WalletInstance of S.WalletUser instructs the TOE to instruct the WSCD to delete the key or key pair*¹³.

Application Note 23

The SFR requires that the WSCD is used to delete keys. Once the operation is completed, the WSCA should ensure that any information, key handle, encrypted key blob, etc. is deleted within the WSCA.

Application Note 24

The ST shall specify the different types of destruction methods of keys linked to S.WalletUser, destructed by WSCD on TOE request, with (iterations of) FCS_CKM.6.

FDP_ACF.1/KeyDeletion Security attribute based access control

FDP_ACF.1.1/KeyDeletion The TSF shall enforce the *KeyDeletion SFP*¹⁴ to objects based on the following:

- (1) *whether the subject is a S.WalletInstance authorised to delete a key or key pair linked to S.WalletUser*
- (2) *[assignment: other rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]*¹⁵.

FDP_ACF.1.2/KeyDeletion The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

- (1) *only an authenticated S.WalletInstance who has been authorized for deletion of a key or key pair linked to S.WalletUser can carry out the delete_key operation*¹⁶.

FDP_ACF.1.3/KeyDeletion The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *[assignment: rules, based on security attributes, that explicitly authorize access of subjects to objects]*.

¹² [assignment: access control SFP]

¹³ [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]

¹⁴ [assignment: access control SFP]

¹⁵ [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]

¹⁶ [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

FDP_ACF.1.4/
KeyDeletion The TSF shall explicitly deny access of subjects to objects based on the following additional rule:

(1) [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*].

Application Note 25

The ST TSS shall briefly describe how the TOE verifies, based on SFP-relevant security attributes, that a S.WalletInstance has been authorized for deletion of a key or a key pair linked to S.WalletUser.

Application Note 26

In the FDP_ACF.1.4, the ST shall explicitly define any applicable rules that deny access of subjects to objects related to actions defined in (iterations of) FIA_AFL.1. E.g. if FIA_AFL.1 defines blocking of a subject, then ST shall state if blocked subject is denied access to objects covert by this access control SFP

FDP_ACC.1/CryptoOperations Subset access control

FDP_ACC.1.1/CryptoOperations

The TSF shall enforce the *CryptoOperations SFP*¹⁷ on:

Subjects: S.WalletUser and S.WalletInstance.

Objects: The assets R.SecretKey linked to S.WalletUser.

Operations: cryptographic_operations: [assignment: list of specific cryptographic operations].

*The S.WalletUser using S.WalletInstance instructs the TOE to instruct the WSCD carry out the operation with the secret key R.SecretKey*¹⁸.

Application Note 27

The ST shall specify the different types of cryptographic operations, which can be carried out by the WSCD on TOE request with (iterations of) FCS_COP.1.

FDP_ACF.1/CryptoOperations Security attribute-based access control

FDP_ACF.1.1/CryptoOperations

The TSF shall enforce the *CryptoOperations SFP*¹⁹ to objects based on the following:

¹⁷ [assignment: *access control SFP*]

¹⁸ [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

¹⁹ [assignment: *access control SFP*]

	(1) <i>whether the subject is a S.WalletUser using S.WalletInstance authorized to carry out the cryptographic operation,</i> (2) <i>whether S.WalletUser is not blocked</i> (3) <i>[assignment: other rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]²⁰.</i>
FDP_ACF.1.2/CryptoOperations	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: (1) <i>only an authenticated S.WalletUser who has been authorised to carry out the cryptographic operation under his sole control²¹.</i>
FDP_ACF.1.3/CryptoOperations	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects].
FDP_ACF.1.4/CryptoOperations	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: (2) <i>S.WalletUser is blocked</i> (3) <i>[assignment: rules, based on security attributes, that explicitly deny access of subjects to objects].</i>

Application Note 28

The ST TSS shall briefly describe how the TOE verifies, based on SFP-relevant security attributes, that the S.WalletUser is authorised to carry out the cryptographic operation and how the sole control is enforced.

Application Note 29

In the FDP_ACF.1.4, the ST shall explicitly define any applicable rules that deny access of subjects to objects related to actions defined in (iterations of) FIA_AFL.1. E.g. if FIA_AFL.1 defines blocking of a subject, then ST shall state if blocked subject is denied access to objects covert by this access control SFP.

The ST may refine this SFR (assignment operation in this FDP_ACF.1.4/CryptoOperations) by adding context for S.WalletUser blocking following FIA_AFL.1/WalletUser.

4.2.4 Identification and authentication (FIA)

FIA_AFL.1/WalletUser Authentication failure handling

²⁰ [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]

²¹ [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

FIA_AFL.1.1/WalletUser	The TSF shall detect when [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]] unsuccessful authentication attempts occur related to <i>S.WalletUser</i> for authentication mechanisms verified by the TSF ²² .
FIA_AFL.1.2/WalletUser	When the defined number of unsuccessful authentication attempts has been met ²³ , the TSF shall <i>block the S.WalletUser</i> and [assignment: list of actions] ²⁴ .

Application Note 30

The failure counter can only be applied for the authentication data, which is verified by the TOE.

Application Note 31

The capability to unblock may be introduced by the ST.

Application Note 32

The ST may refine this SFR (assignment operation in this SFR) by adding context for authentication events for and blocking action (e.g. associate them with R.SecretKey or knowledge factor of R.UserAuthenticationData).

FIA_AFL.1/WalletInstance Authentication failure handling

FIA_AFL.1.1/WalletInstance	The TSF shall detect when [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]] unsuccessful authentication attempts occur related to <i>S.WalletInstance</i> for authentication mechanisms verified by the TSF ²⁵ .
FIA_AFL.1.2/WalletInstance	When the defined number of unsuccessful authentication attempts has been met ²⁶ , the TSF shall [assignment: list of actions].

FIA_AFL.1/WalletProvider Authentication failure handling

²² [assignment: list of authentication events]

²³ [selection: *met, surpassed*]

²⁴ [assignment: list of actions]

²⁵ [assignment: list of authentication events]

²⁶ [selection: *met, surpassed*]

FIA_AFL.1.1/WalletProvider	The TSF shall detect when [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]] unsuccessful authentication attempts occur related to S.WalletProvider for authentication mechanisms verified by the TSF ²⁷ .
FIA_AFL.1.2/WalletProvider	When the defined number of unsuccessful authentication attempts has been met ²⁸ , the TSF shall [assignment: list of actions].

Application Note 33

This SFR expresses the requirement related to S.WalletProvider (both Wallet Provider backend application and Administrator).

If the TSF does not authenticate Wallet Provider Administrator itself, but relies on environment than:

- *A.WalletProvider_Protection shall be refined accordingly, and*
- *this SFR is considered satisfied in relation to the Wallet Provider Administrator.*

This SFR shall be iterated if TSF handles authentication failures of Wallet Provider backend application and Administrator in different ways.

FIA_ATD.1 User attribute definition

FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: <i>the security attributes as defined in FIA_USB.1(FIA_USB.1/User, FIA_USB.1/WalletInstance and FIA_USB.1/WalletProvider)</i> ²⁹ .
-------------	---

FIA_USB.1/WalletUser User-subject binding

FIA_USB.1.1/WalletUser	The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: (1) <i>R.UserAuthenticationData</i> (2) <i>R.SecretKey, R.PublicKey</i> (3) <i>R.WSCDAuthenticationData-WalletUserElement</i> (4) <i>whether S.WalletUser is blocked.</i> (5) <i>[assignment: list of user security attributes]</i>³⁰ to S.WalletUser.
------------------------	--

²⁷ [assignment: list of authentication events]

²⁸ [selection: met, surpassed]

²⁹ [assignment: list of security attributes]

³⁰ [assignment: list of user security attributes]

FIA_USB.1.2/WalletUser	The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: (1) <i>S.WalletUser is not blocked</i> (2) <i>[assignment: rules for the initial association of attributes]</i>.
FIA_USB.1.3/WalletUser	The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on behalf of users: (1) <i>[assignment: rules for the changing of attributes]</i>.

Application Note 34

The R.WSCDAuthenticationData assigned to S.WalletUser relates to data elements which are unique for a user or specific key.

Application Note 35

The ST may refine this SFR (assignment operations) by adding context for S.WalletUser blocking following FIA_AFL.1/WalletUser.

FIA_USB.1/WalletInstance	User-subject binding
--------------------------	----------------------

FIA_USB.1.1/WalletInstance	The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: (1) <i>R.WalletInstanceAuthenticationData</i> (2) <i>[assignment: list of user security attributes]</i> ³¹ to S.WalletInstance.
FIA_USB.1.2/WalletInstance	The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: (1) <i>[assignment: rules for the initial association of attributes]</i>.
FIA_USB.1.3/WalletInstance	The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on behalf of users: (1) <i>[assignment: rules for the changing of attributes]</i>.

FIA_USB.1/WalletProvider	User-subject binding
--------------------------	----------------------

³¹ *[assignment: list of user security attributes]*

FIA_USB.1.1/WalletProvider The TSF shall associate the following user security attributes with subjects acting on the behalf of that user:

- (1) *R.WalletProviderAuthenticationData*
- (2) *R.WSCDAuthenticationData-WalletProviderElement*
- (3) *[assignment: list of user security attributes]*³²

to S.WalletProvider.

FIA_USB.1.2/WalletProvider The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users:

- (1) *[assignment: rules for the initial association of attributes].*

FIA_USB.1.3/WalletProvider The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users:

- (1) *[assignment: rules for the changing of attributes].*

Application Note 36

The R.WSCDAuthenticationData assigned to S.WalletProvider relates to data elements which are related to TOE authentication in WSCD (not unique for a user or specific user key).

FIA_UAU.1 Timing of authentication

FIA_UAU.1.1 The TSF shall allow *[assignment: list of TSF mediated actions]* on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Application Note 37

This SFR shall be iterated if TSF allows different actions for different subjects (S.WalletProvider, S.WalletInstance, S.WalletUser).

In relation to S.WalletProvider two roles shall be considered (backend application and administrator) and handled following approach in Application Note 33.

FIA_UAU.4 Single-use authentication mechanisms

³² *[assignment: list of user security attributes]*

Hierarchical to: No other components.

Dependencies: No other components.

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [assignment: *identified authentication mechanism(s) for at least S.WalletProvider and S.WalletInstance*]³³.

FIA_UAU.5 Multiple authentication mechanisms

FIA_UAU.5.1 The TSF shall provide [selection: [assignment: *list of authentication mechanisms verified by the TOE*], [assignment: *list of authentication mechanisms verified outside the TOE*]]³⁴ to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the [selection; [assignment: *rules describing how the authentication mechanisms verified by the TOE provide authentication*], [assignment: *rules describing how the authentication mechanisms verified outside the TOE provide authentication*]]³⁵.

Application Note 38

This SFR requires the ST author to describe how a user is authenticated by the TOE. If one authentication factor is verified outside of the TOE, the ST TSS shall describe how the TOE verifies that one external authentication factor has been verified outside the TOE, i.e. what kind of evidence is used by the TOE for this verification. At least one authentication factor must be verified by the TOE.

Application Note 39

Authentication of S.WalletProvider or S.WalletInstance may be carried out using one authentication factor.

FIA_UID.1 Timing of identification

Hierarchical to: No other components.

Dependencies: No dependencies.

³³ [assignment: *identified authentication mechanism(s)*].

³⁴ [assignment: *list of multiple authentication mechanisms*]

³⁵ [assignment: *rules describing how the multiple authentication mechanisms provide authentication*]

FIA_UID.1.1	The TSF shall allow [assignment: list of TSF-mediated actions] on behalf of the user to be performed before the user is identified.
FIA_UID.1.2	The TSF shall require each user to be successfully identified before allowing any TSF mediated actions on behalf of that user.

Application Note 40

This SFR applies to S.WalletProvider, S.WalletInstance. The identification of S.WalletUser follows TO_BE_ADDRESSED_WITH_OE.

This SFR shall be iterated if TSF allows different actions for different subjects (S.WalletProvider, S.WalletInstance).

In relation to S.WalletProvider two roles shall be considered (backend application and administrator) and handled following approach in Application Note 33.

4.2.5 Security Management (FMT)

FMT_MSA.1/WalletUser Management of security attributes
--

FMT_MSA.1.1/WalletUser	(1) The TSF shall enforce the <i>User_creation SFP</i> ³⁶ to restrict the ability to create ³⁷ the security attributes <i>S.WalletUser's attributes</i> ³⁸ to <i>S.WalletInstance</i> ³⁹ .
------------------------	--

Application Note 41

The ST TSS shall briefly describe the applicable S.WalletUser's attributes

FMT_MSA.1/Key Management of security attributes

FMT_MSA.1.1/Key	The TSF shall enforce the <i>KeyCreation SFP</i> and <i>KeyDeletion SFP</i> ⁴⁰ to restrict the ability to <i>create and delete</i> ⁴¹ the security attributes
-----------------	---

³⁶ [assignment: *access control SFP(s)*, *information flow control SFP(s)*]

³⁷ [selection: *change_default*, *query*, *modify*, *delete*, [assignment: *other operations*]]

³⁸ [assignment: *list of security attributes*]

³⁹ [assignment: *the authorized identified roles*]

⁴⁰ [assignment: *access control SFP(s)*, *information flow control SFP(s)*]

⁴¹ [selection: *change_default*, *query*, *modify*, *delete*, [assignment: *other operations*]]

R.SecretKey's attributes and R.PublicKey's attributes⁴² to S.WalletInstance and S.WalletUser⁴³.

Application Note 42

The ST TSS shall briefly describe the applicable R.SecretKey's attributes and R.PublicKey's attributes.

FMT_MSA.2 Secure security attributes

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for *all security attributes listed in FIA_USB.1/WalletUser and [assignment: list of security attributes]⁴⁴.*

FMT_MSA.3/WalletUser Static attribute initialization

FMT_MSA.3.1 The TSF shall enforce the *UserCreation SFP⁴⁵* to provide *restrictive⁴⁶* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the *S.WalletInstance⁴⁷* to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.3/Key Static attribute initialization

FMT_MSA.3.1 The TSF shall enforce the *KeyCreation SFP⁴⁸* to provide *restrictive⁴⁹* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the *S.WalletInstance and S.WalletUser⁵⁰* to specify alternative initial values to override the default values when an object or information is created.

⁴² [assignment: *list of security attributes*]

⁴³ [assignment: *the authorized identified roles*]

⁴⁴ [assignment: *list of security attributes*]

⁴⁵ [assignment: *access control SFP, information flow control SFP*]

⁴⁶ [selection, choose one of: *restrictive, permissive, [assignment: other property]*]

⁴⁷ [assignment: *the authorized identified roles*]

⁴⁸ [assignment: *access control SFP, information flow control SFP*]

⁴⁹ [selection, choose one of: *restrictive, permissive, [assignment: other property]*]

⁵⁰ [assignment: *the authorized identified roles*]

FMT_MTD.1/WalletProvider Management of TSF data

FMT_MTD.1 The TSF shall restrict the ability to *modify*⁵¹ the

- (1) *R.WalletProviderAuthenticationData*
- (2) *R.WalletInstanceAuthenticationData*
- (3) *R.WSCDAuthenticationData-WalletProviderElement*
- (4) *[assignment: list of TSF data]*⁵²

to S.WalletProvider⁵³.

Application Note 43

The R.WSCDAuthenticationData data elements restricted to be managed by S.WalletProvider consist of elements for mutual identification and authentication between WSCA and WSCD.

The WalletInstanceAuthenticationData data elements restricted to be managed by S.WalletProvider include at least required a priori data to authenticate S.WalletInstance for the FDP_ACC.1/UserCreation and FDP_ACF.1/UserCreation.

FMT_MTD.1/WalletUser Management of TSF data

FMT_MTD.1 The TSF shall restrict the ability to *modify*⁵⁴ the

- (1) *R.UserAuthenticationData*

to S.WalletUser⁵⁵.

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

- (1) management of the TSF data as defined in FMT_MTD.1/WalletProvider

⁵¹ [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]

⁵² [assignment: *list of TSF data*]

⁵³ [assignment: *the authorized identified roles*]

⁵⁴ [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]

⁵⁵ [assignment: *the authorized identified roles*]

- (2) management of R.UserAuthenticationData as defined in FMT_MTD.1/WalletUser
- (3) management of the S.WalletUser's attributes as defined in FMT_MSA.1/WalletUser
- (4) management of the R.SecretKey's attributes and R.PublicKey's attributes as FMT_MSA.1/Key
- (5) [assignment: list of other management functions to be provided by the TSF]⁵⁶.

FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles

- (1) S.WalletUser
- (2) S.WalletProvider
- (3) S.WalletInstance⁵⁷.

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

4.2.6 Protection of the TSF (FPT)

FPT_RPL.1 Replay detection

FPT_RPL.1.1 The TSF shall detect replay for the following entities: *all operations identified in FDP_ACC.1 (FDP_ACC.1/UserCreation, FDP_ACC.1/KeyCreation, FDP_ACC.1/KeyDeletion, FDP_ACC.1/CryptoOperations) and [assignment: list of identified entities]*.

FPT_RPL.1.2 The TSF shall perform *deny replayed operation and [assignment: list of specific actions]* when replay is detected.

FPT_STM.1 Reliable time stamps

Hierarchical to: No other components.

Dependencies: No dependencies.

⁵⁶ [assignment: list of management functions to be provided by the TSF]

⁵⁷ [assignment: the authorized identified roles]

The TSF shall be able to provide reliable time stamps.

4.2.7 Trusted path/channels (FTP)

FTP_ITC.1/WalletInstance Trusted channel	
FTP_ITC.1.1/WalletInstance	The TSF shall provide a communication channel between itself and another trusted IT product a trusted Wallet Instance that is logically distinct from other communication channels and provides assured identification and authentication of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2/WalletInstance	The TSF shall permit <i>S.WalletInstance</i> ⁵⁸ to initiate communication via the trusted channel.
FTP_ITC.1.3/WalletInstance	The TSF shall initiate communication via the trusted channel for <i>all communication with S.WalletInstance, including</i> (1) <i>create_new_user</i> (2) <i>create_new_key</i> (3) <i>delete_key</i> (4) <i>cryptographic_operations</i> (5) <i>[assignment: list of functions for which a trusted channel is required]</i>⁵⁹.

Application Note 44

The Wallet Instance authentication is handled by FIA_AFL.1/WalletInstance and FIA_UAU.1.

Application Note 45

The ST TSS shall briefly describe what means TOE provides to Wallet Instance to be authenticated.

FTP_ITC.1/WSCD	Trusted channel
FTP_ITC.1.1/WSCD	The TSF shall provide a communication channel between itself and another trusted IT product a trusted certified WSCD that is logically distinct from other communication channels and provides assured identification and authentication of its end points and protection of the channel data from modification or disclosure.

⁵⁸ [selection: the TSF, another trusted IT product]

⁵⁹ [assignment: *list of functions for which a trusted channel is required*]

FTP_ITC.1.2/WSCD

The TSF shall permit **TSF⁶⁰** to initiate communication via the trusted channel.

FTP_ITC.1.3/WSCD

The TSF shall initiate communication via the trusted channel for *all communication with WSCD including*

- (1) *create_new_key*
- (2) *delete_key*
- (3) *cryptographic_operations*
- (4) *[assignment: list of functions for which a trusted channel is required]⁶¹.*

Application Note 46

The ST TSS shall briefly describe how mutual authentication is achieved.

4.3 Security assurance requirements

Assurance Class	Assurance Components
Security Target (ASE)	ST introduction (ASE_INT.1)
	Conformance claims (ASE_CCL.1)
	Security problem definition (ASE_SPD.1)
	Security objectives (ASE_OBJ.2)
	Extended components definition (ASE_ECD.1)
	Derived security requirements (ASE_REQ.2)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Security architecture description (ADV_ARC.1)
	Complete functional specification (ADV_FSP.4)
	Basic modular design (ADV_TDS.3)
	Implementation representation of the TSF (ADV_IMP.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)

⁶⁰ [selection: *the TSF, another trusted IT product*]

⁶¹ [assignment: *list of functions for which a trusted channel is required*]

Life cycle support (ALC)	Production support, acceptance procedures and automation (ALC_CMC.4)
	Problem tracking CM coverage (ALC_CMS.4)
	Delivery procedures (ALC_DEL.1)
	Identification of security measures (ALC_DVS.1)
	Developer defined life-cycle model (ALC_LCD.1)
	Well-defined development tools (ALC_TAT.1)
	Flaw reporting procedures (ALC_FLR.2)
Tests (ATE)	Functional testing (ATE_FUN.1)
	Analysis of coverage (ATE_COV.2)
	Testing: basic design (ATE_DPT.1)
	Independent testing – sample (ATE_IND.2)
Vulnerability assessment (AVA)	Advanced methodical vulnerability analysis (AVA_VAN.5)

Table 2 Security Assurance Requirements

4.4 Security requirements rationale

4.4.1 Coverage

	OT.Data_Protection	OT.Algorithms	OT.Authentication	OT.Communication_Protection
FAU_GEN.1	X			
FAU_GEN.2	X			
FCS_CKM.1		X		
FCS_CKM.6		X		

FCS_COP.1		X		
FDP_ACC.1/UserCreation	X		X	
FDP_ACF.1/UserCreation	X		X	
FDP_ACC.1/KeyCreation	X	X	X	
FDP_ACF.1/KeyCreation	X	X	X	
FDP_ACC.1/KeyDeletion	X	X	X	
FDP_ACF.1/KeyDeletion	X	X	X	
FDP_ACC.1/CryptoOperations	X	X	X	
FDP_ACF.1/CryptoOperations	X	X	X	
FIA_AFL.1/WalletUser			X	
FIA_AFL.1/WalletInstance			X	X
FIA_AFL.1/WalletProvider			X	
FIA_ATD.1			X	
FIA_USB.1/WalletUser			X	
FIA_USB.1/WalletInstance			X	
FIA_USB.1/WalletProvider			X	
FIA_UAU.1			X	X
FIA_UAU.4			X	
FIA_UAU.5			X	
FIA_UID.1			X	
FMT_MSA.1/WalletUser	X			
FMT_MSA.1/Key	X			
FMT_MSA.2	X			
FMT_MSA.3/WalletUser	X			
FMT_MSA.3/Key	X			
FMT_MTD.1/WalletProvider	X			
FMT_MTD.1/WalletUser	X			
FMT_SMF.1	X			

FMT_SMR.1	X			
FPT_RPL.1				X
FPT_STM.1	X			
FTP_ITC.1/WalletInstance				X
FTP_ITC.1/WSCD				X

Table 3 TOE Security Objective mapping to SFRs

4.4.2 Rationale

OT.Data_Protection is covered by the requirements in:

- FMT_SMR.1 specifying user roles recognized by TOE;
- FDP_ACC.1/UserCreation and FDP_ACF.1/UserCreation requiring access control rules for Wallet User creation in TOE;
- FDP_ACC.1/KeyCreation, FDP_ACF.1/KeyCreation, FDP_ACC.1/KeyDeletion, FDP_ACF.1/KeyDeletion, FDP_ACC.1/CryptoOperations and FDP_ACF.1/CryptoOperations enforcing access control for all operations linked to these users;
- FMT_SMF.1 specifying management function related to security attributes and other TSF data
- FMT_MSA.1/WalletUser, FMT_MSA.1/Key, FMT_MSA.2, FMT_MSA.3/WalletUser, FMT_MSA.3/Key restricting access to the management of security attributes related to Wallet Users and theirs' keys,
- FMT_MSA.2 enforcing secure values of all the Wallet User security attributes;
- FMT_MTD.1/WalletProvider and FMT_MTD.1/WalletUser restricting access to TSF data;
- FAU_GEN.1, FAU_GEN.2 and FPT_STM.1 enforcing audit record generation with user identity association and reliable time stamps.

OT.Algorithms is covered by the requirements in:

- FDP_ACC.1/KeyCreation, FDP_ACF.1/KeyCreation, FDP_ACC.1/KeyDeletion, FDP_ACF.1/KeyDeletion, FDP_ACC.1/CryptoOperations and FDP_ACF.1/CryptoOperations enforcing secure generation, usage and destruction of keys using WSCD; and
- FAU_GEN.1, FCS_CKM.1, FCS_CKM.6, FCS_COP.1 concerning cryptographic algorithms (including algorithm parameters and key lengths) used by TOE for TSF protection and selected by the TOE and used by the WSCD.

OT.Authentication is covered by the requirements in:

- FIA_ATD.1, FIA_USB.1/WalletUser, FIA_USB.1/Wallet, FIA_USB.1/WalletProvider enforcing binding of security attributes maintained by TSF to subjects
- FIA_UID.1 enforcing identification of all subjects;
- FIA_UAU.1, FIA_UAU.4 FIA_UAU.5 enforcing authentication of all subjects including f two factor authentication of Wallet User, and single-use authentication mechanisms for Wallet User and Wallet Instance;s
- FIA_AFL.1/WalletUser, FIA_AFL.1/WalletInstance, FIA_AFL.1/WalletProvider enforcing rules for authentication failure handling for all the subjects, including blocking the S.WalletUser after a defined number of unsuccessful attempts

- FDP_ACC.1/UserCreation, FDP_ACF.1/UserCreation, FDP_ACC.1/KeyCreation, FDP_ACF.1/KeyCreation, FDP_ACC.1/KeyDeletion, FDP_ACF.1/KeyDeletion, FDP_ACC.1/CryptoOperations and FDP_ACF.1/CryptoOperations enforcing authentication of all subjects before all operations (user creation, key creation, key deletion, key usage – cryptographic operations) and preventing operations for blocked subjects.

OT.Communication_Protection is covered by the requirements in:

- FTP_ITC.1/WalletInstance which require trusted channel with mutual authentication between Wallet Instance and TOE;
- FIA_AFL.1/WalletInstance and FIA_UAU.1 which handle details of Wallet Instance authentication by TOE
- FTP_ITC.1/WSCD which require trusted channel with mutual authentication between TOE and WSCD;
- FPT_RPL.1 ensures protection against replay for all operations (user creation, key creation, key deletion, key usage – cryptographic_operations).

4.4.3 SFR Dependencies

Requirement	Dependencies	Fulfilled by
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FAU_GEN.2	FAU_GEN.1 FIA_UID.1	FAU_GEN.1 FIA_UID.1
FCS_CKM.1	[FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1] FCS_CKM.3 [FCS_RBG.1 or FCS_RNG.1] FCS_CKM.6	Application Note 47 <i>The ST shall cover the dependencies of each iteration of this SFR accordingly (TOE implementation dependent).</i>
FCS_CKM.6	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	Application Note 48 <i>The ST shall cover the dependencies of each iteration of this SFR accordingly (TOE implementation dependent).</i>
FCS_COP.1	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5s] FCS_CKM.3	Application Note 49 <i>The ST shall cover the dependencies of each iteration of this SFR accordingly (TOE implementation dependent).</i>

FDP_ACC.1/UserCreation	FDP_ACF.1	FDP_ACF.1/UserCreation
FDP_ACF.1/UserCreation	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/UserCreation FMT_MSA.3/WalletUser
FDP_ACC.1/KeyCreation	FDP_ACF.1	FDP_ACF.1/KeyCreation
FDP_ACF.1/KeyCreation	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/KeyCreation FMT_MSA.3/Key
FDP_ACC.1/KeyDeletion	FDP_ACF.1	FDP_ACF.1/KeyDeletion
FDP_ACF.1/KeyDeletion	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/KeyDeletion FMT_MSA.3/Key
FDP_ACC.1/CryptoOperations	FDP_ACF.1	FDP_ACF.1/CryptoOperations
FDP_ACF.1/CryptoOperations	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/CryptoOperations FMT_MSA.3/Key
FIA_AFL.1/WalletUser	FIA_UAU.1	FIA_UAU.1
FIA_AFL.1/WalletInstace	FIA_UAU.1	FIA_UAU.1
FIA_AFL.1/WalletProvider	FIA_UAU.1	FIA_UAU.1
FIA_ATD.1	No dependencies	
FIA_USB.1/WalletUser	FIA_ATD.1	FIA_ATD.1
FIA_USB.1/WalletInstance	FIA_ATD.1	FIA_ATD.1
FIA_USB.1/WalletProvider	FIA_ATD.1	FIA_ATD.1
FIA_UAU.1	FIA_UID.1	FIA_UID.1
FIA_UAU.5	No dependencies	
FIA_UID.1	No dependencies	
FMT_MSA.1/WalletUser	[FDP_ACC.1, or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1/UserCreation FMT_SMR.1 FMT_SMF.1
FMT_MSA.1/Key	[FDP_ACC.1, or FDP_IFC.1] FMT_SMR.1	FDP_ACC.1/KeyCreation and FDP_ACC.1/KeyDeletion FMT_SMR.1

	FMT_SMF.1	FMT_SMF.1
FMT_MSA.2	[FDP_ACC.1, or FDP_IFC.1] FMT_MSA.1 FMT_SMR.1	FDP_ACC.1/UserCreation and FDP_ACC.1/KeyCreation FMT_MSA.1/WalletUser and FMT_MSA.1/Key FMT_SMR.1
FMT_MSA.3/WalletUser	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1/WalletUser FMT_SMR.1
FMT_MSA.3/Key	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1/Key FMT_SMR.1
FMT_MTD.1/WalletProvider	FMT_SMR.1 FMT_SMF.1	FMT_SMR.1 FMT_SMF.1
FMT_MTD.1/WalletUser	FMT_SMR.1 FMT_SMF.1	FMT_SMR.1 FMT_SMF.1
FMT_SMF.1	No dependencies	
FMT_SMR.1	FIA_UID.1	FIA_UID.1
FPT_RPL.1	No dependencies	
FPT_STM.1	No dependencies	
FTP_ITC.1/WalletInstance	No dependencies	
FTP_ITC.1/WSCD	No dependencies	

Table 4 SFR Dependencies Rationale

Bibliography

[CC1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, CC:2022, Revision 1, CCMB-2022-11-001, November 2022.
[CC2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, CC:2022, Revision 1, CCMB-2022-11-002, November 2022.
[CC3]	Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, CC:2022, Revision 1, CCMB-2022-11-003, November 2022.
[Errata]	Title: Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), version 1.2.
[ARF]	European Digital Identity Wallet Architecture and Reference Framework, EU Member States experts. The [ARF] is available at: https://github.com/eu-digital-identity-wallet/eudi-doc-architecture-and-reference-framework/tags Current version is 2.8.0.
[IA5a]	[IA5a] COMMISSION IMPLEMENTING REGULATION (EU) 2024/2979 of 28 November 2024 laying down rules for the application of Regulation (EU) No 910/2014 of the European Parliament and of the Council as regards the integrity and core functionalities of European Digital Identity Wallets.
[IA5c]	COMMISSION IMPLEMENTING REGULATION (EU) 2024/2981 of 28 November 2024 laying down rules for the application of Regulation (EU) No 910/2014 of the European Parliament and the Council as regards the certification of European Digital Identity Wallets.
[CIR 2015/1502]	COMMISSION IMPLEMENTING REGULATION (EU) 2015/1502 of 8 September 2015 on setting out minimum technical specifications and procedures for assurance levels for electronic identification means pursuant to Article 8(3) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market.
[eIDAS]	Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC with the amendments in Regulation (EU) 2024/1183 of the European Parliament and of the Council of 11 April 2024 amending Regulation (EU) No 910/2014 as regards establishing the European Digital Identity Framework.

[EUCC ACM]	European Cybersecurity Certification Group, Sub-group on Cryptography, Agreed Cryptographic Mechanisms, ENISA.
[CEN EN PP 419 221-5]	Protection profiles for TSP Cryptographic modules - Part 5 Cryptographic Module for Trust Services, CEN, 2016.