EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL

Ensuring High Assurance of EUDI Wallets
& the NL National eID Scheme



NL-WSCA

NL EUDI Wallet Certification System

Protection Profile for WSCA in a remote HSM architecture

RC1.1

2026.10.29

Document reference	NL-WSCA
Version	RC1.1
Status	Release Candidate
Date	29/10/2026
Document type	Protection Profile for WSCA in a remote HSM architecture
Parent framework	NL-CS (parts1/2/3/4)
Scheme Owner	Kingdom of the Netherlands Minister of Economic Affairs
Certification target	European Digital Identity Wallets
Primary audience	Scheme Owners, Evaluation Laboratories; CABs, EUDI Wallet providers
Secondary audience	EUDI Wallet ecosystem stakeholders
Assurance level <identity proofing CIR 2015/1502> and <evaluation, EU Regulation 2019/881>	High/High
Related Documents	NL-WSCA-conditions NL-WI (-conditions)

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Version	Change	State	Date
RC1.0	Result of European Collaboration / Discussions	Release Candidate	2026-10-27
RC1.1	Release Candidate for publication following review round NL Certification System Mngt Board	Release Candidate	2026-10-31

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Background

This Protection Profile (PP) is developed based on Regulation (EU) No 910/2014, as amended by Regulation (EU) 2024/1183 (eIDAS 2.0), and Commission Implementing Regulation (EU) 2024/2981.

The architectural configuration described in this PP, hereafter referred to as the **Remote HSM architecture**, is one of the deployment models contemplated by Commission Implementing Regulation (EU) 2024/2981 for the certification of European Digital Identity Wallets. Article 2(4) of that Regulation defines the WSCA as an application that manages critical assets by being linked to and using the cryptographic and non-cryptographic functions provided by the WSCD. In the Remote HSM architecture, the WSCA executes on Wallet Provider backend infrastructure (**WSCA-BE**), while the WSCD is implemented as a certified remote HSM. The HSM manages the critical private keys of the Wallet Unit, including the attested keys for the KA and, if applicable, the WIA device-binding key pair. The Wallet Instance additionally holds a device-resident key in hardware-backed storage (e.g., a Secure Element or platform-provided hardware-backed keystore), which serves as the possession-based authentication factor for remote authentication to the WSCA/WSCD.

This PP applies to the WSCA-BE as a standalone **Target of Evaluation (TOE)** for Common Criteria evaluation under the EUCC scheme established by Commission Implementing Regulation (EU) 2024/482. A certificate of conformity issued against this PP may serve as an assurance input for national wallet certification schemes pursuant to Article 4 of Commission Implementing Regulation (EU) 2024/2981. The TOE of this PP will require a secure and certified WSCD as well as a secured underlying server and secure operational environment as defined by the objectives in chapter 4.

Terminology

This Protection Profile uses the terminology defined in the **European Digital Identity Wallet Architecture and Reference Framework (ARF)** and the applicable EU regulations. Unless otherwise stated, terms such as **Wallet Unit, Wallet Instance, Wallet Secure Cryptographic Application (WSCA), Wallet Secure Cryptographic Device (WSCD), Wallet Provider, Wallet User, Wallet Instance Attestation (WIA), Key Attestation (KA),** and **critical assets** shall have the meaning defined in the ARF and the applicable Commission Implementing Regulations.

Where this Protection Profile introduces terms specific to the **Remote HSM architecture**, these terms are explicitly defined in this Protection Profile. In case of ambiguity, the definitions in the applicable EU legislation take precedence over terminology introduced in this Protection Profile. The following terms defining the secure authentication protocol between the Wallet Instance and the WSCA/WSCD are specific to the **Remote HSM architecture**:

- **Remote User Authentication Protocol (RUAP):** The protocol enforces that every PIN verification is performed through a live and fresh interaction between the Wallet Instance and the WSCA. This ensures that the confidentiality of the PIN is preserved, while verification is subject to controlled and rate limited processing at the WSCA. The Wallet Instance must demonstrate the User Authentication Evidence (UAE), which is derived both from a knowledge-based factor (derived from the PIN) and a possession-based factor. The User Authentication Evidence (UAE) is verified against the Authentication Reference Data (ARD) which is shared during enrolment.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

- **Authentication Reference Data (ARD):** Authentication Reference Data is shared with the WSCA/WSCD during enrolment and used at the WSCA/WSCD to verify the Authentication Reference Data (UAE).
- **User Authentication Evidence (UAE)** Cryptographic evidence bound to a user authentication event at the wallet instance, provided over the Remote Authentication Channel and verified by the WSCA before it performs a wallet cryptographic operation.

Relationship to Other Standards and Schemes

This Protection Profile is intended to be used within the legal, regulatory, certification and technical framework established by the following documents and standards.

- **eIDAS 2.0:** Regulation (EU) No 910/2014, as amended by Regulation (EU) 2024/1183, establishing the European Digital Identity Framework.
- **EUDI Wallet Certification:** Commission Implementing Regulation (EU) 2024/2981, laying down rules for the certification of European Digital Identity Wallets.
- **EUCC Scheme:** Commission Implementing Regulation (EU) 2024/482 establishing the European Common Criteria-based cybersecurity certification scheme (EUCC), including its applicable amendments and supporting documents.
- **EUDI Wallet Core Functionality and Integrity:** Commission Implementing Regulation (EU) 2024/2979.
- **EUDI Wallet Protocols and Interfaces:** Commission Implementing Regulation (EU) 2024/2982.
- **PID and EAA:** Commission Implementing Regulation (EU) 2024/2977, laying down rules concerning person identification data and electronic attestations of attributes.
- **Applicable Standards and Specifications:** Commission Implementing Regulation (EU) **2026/1731**, amending Regulations (EU) 2024/2977, 2024/2979, 2024/2980 and 2024/2982 with updated standards and specifications.
- **Wallet-Relying Parties:** Commission Implementing Regulation (EU) 2025/848 concerning the registration of wallet-relying parties and their access certificates.
- **Certified Wallets:** Commission Implementing Regulation (EU) 2025/849 concerning the submission of information to the Commission and the Cooperation Group for the list of certified EUDI Wallets.
- **Assurance Levels for Electronic Identification:** Commission Implementing Regulation (EU) 2015/1502.
- **EUDI Wallet Architecture and Reference Framework (ARF):** The current applicable version of the ARF, including its requirements, definitions, technical specifications, rulebooks and certification requirements. The ARF provides the technical architecture and reference specifications supporting the EUDI Wallet ecosystem.
- **ECCG Agreed Cryptographic Mechanisms (ACM):** The current applicable version.
- **Common Criteria:** Latest version published on the CCRA publications page, as applicable through the EUCC scheme.
- **Cryptographic and PKI standards:** Applicable ETSI, ISO/IEC and IETF standards referenced by the ARF and the applicable implementing regulations.
- **CEN Protection Profile for TSP Cryptographic Modules:** EN 419 221-5:2018, Protection Profiles for TSP Cryptographic Modules – Part 5: Cryptographic Module for Trust Services. Applicable to the cryptographic functionality and key protection provided by the WSCD; the WSCD may claim conformance to this Protection Profile.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Table of Contents

Background 4
Terminology 4
Relationship to Other Standards and Schemes 5
1 Introduction 7
1.1 Protection Profile Reference Identification 7
1.2 TOE Overview 7
1.3 Wallet Unit Lifecycle and Cryptographic Trust Relationships 12
2 Conformance Claims 20
2.1 Common Criteria Conformance Claim 20
2.2 Package Claim 20
2.3 Protection Profile Conformance Claim 20
2.4 Conformance Statement 20
2.5 Conformance claim rationale 20
3 Security Problem Definition 21
3.1 Assets 21
3.2 Subjects 23
3.3 Threats 23
3.4 Organisational Security Policies 25
3.5 Assumptions 26
4 Security Objectives 28
4.1 Security Objectives for the TOE 28
4.2 Security Objectives for the Operational Environment 30
5 Extended components definition 32
5.1 Protected Audit Event Storage (FAU_STG_EXT) 32
5.2 Authentication protocol with transaction binding FIA_UAE_EXT 33
6 Security Functional Requirements 35
6.1 Class FAU — Security Audit 35
6.2 Class FCO — Communication 36
6.3 Class FCS — Cryptographic Support 37
6.4 Class FDP — User Data Protection 38
6.5 Class FIA — Identification and Authentication 40
6.6 Class FMT — Security Management 43
6.7 Class FPT — Protection of the TSF 45
6.8 Class FTP — Trusted Path/Channel 46
7 Security Assurance Requirements 49
8 Requirement Objective and Dependency Rationale 51
8.1 Security Objectives Rationale 51
8.2 Security Functional Requirements Rationale 56
8.3 Security Dependency Rationale 61
8.4 Security Assurance Requirements Rationale 65
A (informative) Mapping to (EU) 2024/2981 Risk Register 66
B (informative) Completeness Analysis of the Security Problem Definition 68
B.1 Coverage of the TOE Security Functionality 68
B.2 Coverage of Critical Assets 69
B.3 Coverage of the Operational Environment 70

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

1 Introduction

1.1 Protection Profile Reference Identification

Title: Protection Profile for WSCA in the remote HSM architectures

PP Identification: CC-PP-WSCA-REMOTE

CC Identification: CC:2022 (Revision 1), ISO/IEC 15408:2022

Version: 0.9

Date: Thursday, 29 October 2026

Status: Not Certified

Assurance Level: EAL4 augmented with AVA_VAN.5, ALC_DVS.2 and ALC_FLR.3

Authors: SGS Brightsight (ITSEF and CB)

1.2 TOE Overview

The Target of Evaluation (TOE) is the Wallet Secure Cryptographic Application (WSCA) deployed in the wallet provider's back-end infrastructure (WSCA-BE (TOE)). The TOE operates as an authenticated remote signing service: it verifies the user's identity and authorisation, validates the wallet instance key binding, and instructs the Wallet Secure Cryptographic Device (WSCD-HSM) to generate signing keys and to sign the data submitted by the Wallet Instance used in the communication with the PID/Attestation providers and the relying parties. The TOE manages the full lifecycle of Attested key pairs (and optionally WIA device-binding key pairs). However, the attested private keys will never leave the security context of the WSCD-HSM. All cryptographic operations are handled by the WSCD-HSM, which resides in the WSCA-BE physical protected environment.

1.2.1 The Wallet Instance (TOE) in the Wallet Ecosystem

The Wallet Secure Cryptographic Application deployed in the backend, WSCA-BE, is one of the components in the **Wallet Unit**, which acts as part of the **Wallet Ecosystem**. In ***Figure*** 1, the main components of the Wallet Unit and Wallet Ecosystem are depicted.

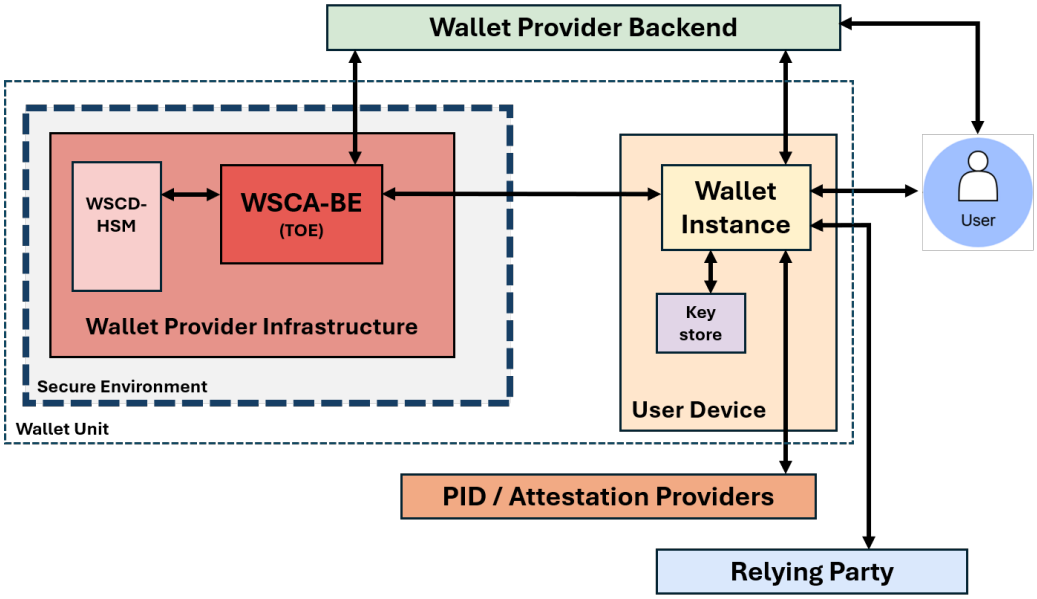


***Figure** 1 — The wallet Unit in the Wallet Ecosystem*

1.2.2 Wallet Unit components

The following components are the core components in the **Wallet Unit**, as depicted in ***Figure*** 1:

User device (UD): The hardware, operating system and software environment on which the Wallet Instance executes. The User Device provides the execution environment, local security mechanisms and a local key store for the Wallet Instance. The local key store may be used to generate, store and use critical cryptographic assets. The security posture of the User Device, including the integrity of its operating system and relevant security mechanisms, contributes to the overall security of the Wallet Unit.

- **Wallet Instance (WI)**: The mobile application installed on the User Device. The Wallet Instance authenticates ecosystem entities, obtains User approval for security-sensitive actions, manages credentials and cryptographic assets, and initiates cryptographic operations. It shall protect sensitive Wallet User information and ensure that security-sensitive operations are performed only under the applicable authorization and User-control conditions. It interfaces with a local **key store** on the user device for the management of critical cryptographic assets.
- **Wallet Secure Cryptographic Application (WSCA-BE) (TOE)**: A WSCA-BE implementation deployed on a hardened backend server. It provides security-critical cryptographic functionality sourced through interfaces with the WSCD. The WSCA-BE enforces the applicable security controls for cryptographic operations and ensures that operations involving critical assets are performed only after the required authorization and User authentication conditions have been satisfied.
- **Wallet Secure Cryptographic Device (WSCD)**: A hardware security device used by the WSCA to protect critical cryptographic assets and perform security-sensitive cryptographic operations. The WSCD shall protect the confidentiality and integrity of protected private keys and shall prevent unauthorized export or use of protected private keys. The WSCD shall be certified against a cryptographic module certified protection profile EN 419 221-5:2018 or a different applicable Protection Profile providing at least equivalent functionality and assurance.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

1.2.3 Wallet Ecosystem Entities

The following entities (depicted in ***Figure*** 1) are part of the **Wallet Ecosystem** and interact with the Wallet Unit:

- **Wallet Provider Backend (WPB):** Provides Wallet Provider services, including Wallet Unit onboarding, activation, management, maintenance and security-status related services. The Wallet Provider is responsible for establishing the required trust relationship with the Wallet Instance, issuing the WIA and KA, and taking appropriate action when the security or trustworthiness of a Wallet Unit is compromised.
- **PID and Attestation Providers:** Providers that issue PIDs and Attestations to the Wallet Unit. Before issuance, the Provider authenticates the Wallet Unit and verifies the applicable WIA and KA, including their validity and revocation status. For device-bound credentials, the Provider verifies proof-of-possession of the private key corresponding to the public key contained in the KA and binds the issued credential to that key.
- **Relying Parties (RPs):** Entities that request and verify PID or Attestation attributes from the Wallet Unit to provide a service to the User. The RP authenticates to the Wallet Unit using the applicable Access Certificate and provides the information required for the User to understand and approve the requested disclosure. The RP is responsible for verifying the authenticity, integrity, freshness and applicable cryptographic binding of the received presentation.
- **Access Certificate Infrastructure**: (not in in ***Figure*** 1): Provides Access Certificates used to authenticate relevant ecosystem entities. The associated trust infrastructure provides the trust anchors required by the Wallet Unit and other ecosystem participants to verify the authenticity of Access Certificates.

1.2.4 Principal security interactions

The principal high level data flows and trust relationships are as follows between the entities:

- **WSCA-BE (TOE) — Wallet Instance (WI):** The Wallet Instance communicates with the WSCA-BE (TOE) through a **mutually authenticated and protected channel** providing confidentiality and integrity. User Authentication and the Remote User Authentication Protocol (RUAP), including User Authentication Evidence (UAE) and Authentication Reference Data (ARD), are used to authenticate the User and authorise security-sensitive operations. The WSCA invokes the WSCD for operations involving protected cryptographic assets.
- **WSCA-BE (TOE) — WSCD:** The WSCA-BE communicates with the WSCD using the secure channel supported by the WSCD. Per-key authentication shall be implemented to ensure that security-sensitive cryptographic operations are authorised before execution by the WSCD. Per-key authentication data shall be securely deleted from the TOE and be managed by the WI and provided the TOE as required for the requested operation.
- **WSCA-BE (TOE) — Wallet Provider Backend (WPB):** The WSCA-BE provides an authentication mechanisms required for the Wallet Provider to authenticate the attested public keys before using them to issue and sign the Key Attestation (KA) and Wallet Instance Attestation (WIA).
  All communication between the WSCA-BE and the WPB shall be conducted over a secure channel providing mutual authentication, confidentiality and integrity.
- **Wallet Instance — Wallet Provider Backend (WPB):** The Wallet Instance communicates with the Wallet Provider for installation, onboarding, activation, Wallet Unit management and security-status-related services. During onboarding and activation, the Wallet Provider receives the attested keys and, where applicable, the

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

WIA device-binding key material, and securely provides the WIA and KA to the Wallet Instance.

- **Wallet Instance — PID/Attestation Providers:** The Wallet Instance communicates with PID and Attestation Providers to request and receive PIDs or Attestations. Where required, the Wallet Unit provides the applicable WIA and KA. The Provider verifies these objects and may request proof of possession of the corresponding private keys before issuing a device-bound credential. The WSCA/WSCD performs the required cryptographic operations and proof-of-possession using the protected private keys.
- **Wallet Instance — Relying Party:** The Wallet Instance participates in credential presentation using the applicable presentation protocol. It authenticates the Relying Party, evaluates the presentation request, obtains explicit User approval and performs selective disclosure. For device-bound credentials, the Wallet Unit uses the applicable WSCA/WSCD to perform the required cryptographic operations and proof-of-possession.

1.2.5 The WSCA-BE (TOE) located inside a Secure Environment

The WSCA-BE (TOE) is deployed on a **dedicated hardened server** that runs no services other than the WSBA-BE. This server is positioned behind **gateway servers** that manages and filters all external communications to the **Wallet Provider backend** and the **Wallet Instance**. This architecture enables the TOE to maintain a minimal and secure design. The WSCA is expected to use an external **database** to securely store the Authentication Reference Data (ARD) for the individual Wallet Units. Every security-relevant action is an audit log entry, which is cryptographically tied to specific wallet units and transactions, and is transmitted to an external **log server**.

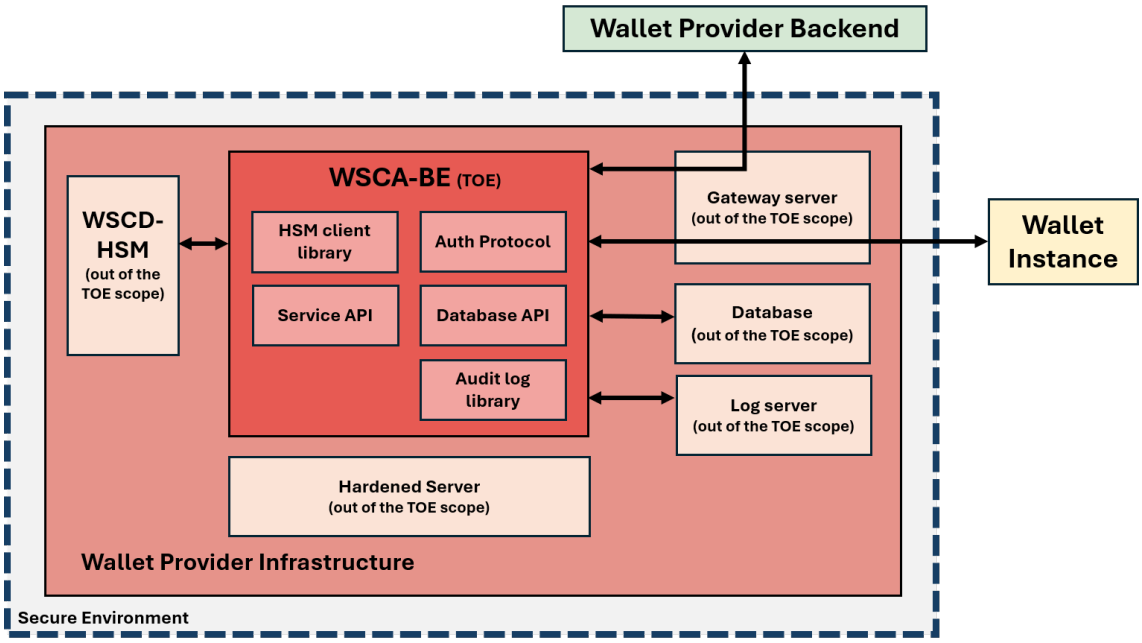


***Figure** 2 — TOE deployment boundary*

The WSCA-BE (TOE), the hardened server, the WSCD, the Gateway server and external databases, the log servers, and their connections, shall operate in a protected environment that limits physical access to the TOE to authorised privileged users.

1.2.5.1 Non-TOE Hardware/Software

Following ***Figure** 2*, the following non-TOE Hardware and/or Software would be required:

- **WSCD-HSM:** An appropriate HSM following CEN EN PP 419 221-5 or a different applicable Protection Profile providing at least equivalent functionality and assurance.
- **Hardened Server:** The TOE underlying hardware is be hardened against remote side-channel attacks and microarchitectural attacks.
- **Gateway Server:** Internet facing gateways shall be used for, at least, DDoS protection, intrusion detection, antivirus functionality and firewall functionality.
- **Database:** External databases used for storage of TOE assets.
- **Log Server:** Receives audit log messages from the TOE. The environment shall ensure that log data is not deleted.
- **Wallet Provider Backend:** The Wallet Provider Backend enables the Wallet Provider to connect to the TOE
- **Wallet Instance:** The Wallet User uses the wallet instance to request operations from the TOE. The wallet instance shall be certified according to (EU) 2024/2981.
- Reliable Time Source: A reliable time source needs to be provided to the TOE.

1.2.6 The WSCA-BE (TOE) major security features

The WSCA-BE (TOE) has the following major security features, as it:

- **Secure communication:** Establishes protected communication channels between the Wallet Instance and the WSCA-BE providing **mutual authentication, confidentiality and integrity**, including protection against replay and message manipulation where required.
- **Remote User Authentication:** Implements the **Remote User Authentication Protocol (RUAP)**, including enrolment and management of Authentication Reference Data (ARD), and verifies transaction-bound and freshly derived **User Authentication Evidence (UAE)** before allowing security-sensitive operations, providing sole control to user specific assets. TOE allows for PIN reset using a PUK mechanism. Per-key authentication data shall be held by the Wallet Instance and be provided at every security-sensitive cryptographic operation of the applicable key, to ensure sole control of the Wallet User before the execution by the WSCD.
- **Wallet Provider authentication and authorization:** Authenticates the **Wallet Provider** and enforces role-based access control so that Wallet Provider functions (e.g. audit data review) and security-sensitive operations can only be performed by an **authenticated and authorized Wallet Provider**.
- **Cryptographic key management:** Manages the lifecycle and metadata of Wallet Unit cryptographic keys, including attested keys and, where applicable, the WIA device-binding key, while delegating private-key protection and cryptographic operations to the **WSCD-HSM**.
- **WSCD-HSM access-control protection:** Protects the **access-control data and authorization information** used to control access to WSCD-HSM services, preventing unauthorized modification or use of this data to obtain unauthorized cryptographic operations.
- **Attestation support:** Supports the Wallet Provider in generating and signing the **Key Attestation (KA)** and **Wallet Instance Attestation (WIA)**. The TOE provides **cryptographic evidence binding the attested public keys to keys generated and managed by the WSCA/WSCD**, enabling the Wallet Provider to verify the authenticity of the public keys before including them in the KA or WIA.
- **Protected cryptographic operations:** Ensures that security-sensitive operations involving protected Wallet Unit assets are performed only following the required **User authentication and authorization**, while private keys remain under WSCD-HSM protection.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

- **Multi-tenant isolation:** Provides isolation between concurrently served Wallet Units, protecting their key metadata, credentials and security-relevant data within the WSCA-BE and maintaining separation between tenants.
- **Security audit:** Generates **tamper-evident audit records** for security-relevant operations, attributable to the relevant Wallet Unit and transaction, and securely transfers them to the external log server.
- **Self protection:** Self testing, secure initalization and preserving of a secure WSCA state is mandated.

The WSCA-BE (TOE) does NOT include the WSCD-HSM hardware or firmware, the Wallet Instance application or device, log server, the Internet-facing gateway or load balancer or any wallet provider backend services other than the WSCA-BE module itself.

1.3 Wallet Unit Lifecycle and Cryptographic Trust Relationships

Note: This section and its subsections are of informative nature, describing the TOE and its place in EUDI Wallet ecosystem. The ST author is not expected to reproduce this section in the ST.

The WSCA-BE (TOE) forms part of a Wallet Unit within the EUDI Wallet ecosystem. This section provides context for the security and trust relationships established during the Wallet Unit lifecycle.

It describes how the WSCA-BE (TOE) interacts with the Wallet Provider, Wallet Instance, keystores, PID and Attestation Providers, and Relying Parties, and how cryptographic mechanisms are used to establish authenticity, integrity, confidentiality and cryptographic binding where required.

The section distinguishes between trust established through cryptographic mechanisms and trust established through certification, registration, trusted lists and other ecosystem governance mechanisms.

1.3.1 Wallet Instance Installation

The Wallet Instance should preferably be distributed through an official application distribution channel of the User Device operating system. If the Wallet Instance is distributed through another mechanism, the Wallet Provider shall provide a mechanism enabling the User to verify the authenticity and integrity of the Wallet Instance before installation.

1.3.1.1 Mutual Authentication Channel

After installation, the Wallet Instance establishes the required secure communication channels with:

- the Wallet Provider; and
- the WSCA/WSCD.

Communication between the Wallet Instance and Wallet Provider shall provide mutual authentication and shall provide confidentiality and integrity protection where required.

1.3.1.2 Initial Data Requested by Wallet Provider

The Wallet Instance will now send its initial data to the Wallet Provider which contains technical data from the Wallet Instance, including cryptographic and hardware capabilities (such as WSCD and keystore details), system diagnostics (including OS, SDK, and library

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

versions, plus information about available communication interfaces such as BLE and NFC), and security posture signals such as rooting, emulator, and malware detection.

The Wallet Provider shall limit collected device and security-posture information to the data necessary for Wallet Solution deployment, maintenance and security-posture monitoring. Collection shall be minimised and performed with appropriate privacy safeguards.

1.3.2 Wallet Activation

For Wallet Activation, the following steps are performed:

- Creation of a Wallet Provider user account and association of the Wallet Instance with that account (further detailed in section 1.3.2.1).
- Registration of the User authentication methods that will be used to access and operate the wallet (further detailed in section 1.3.2.2).
- Initial Generation and issuance of the WIA and KA (further detailed in section 1.3.2.3)

During activation, the Wallet Provider establishes the required trust relationship with the Wallet Instance and provides the WIA and KA required for subsequent Wallet Unit operations. The WIA provides information about the Wallet Instance and Wallet Solution, while the KA provides information about the WSCA/WSCD or applicable keystore and the public keys used by the Wallet Unit.

The User authentication mechanisms established during activation are subsequently used to control access to the Wallet Unit and, where applicable, to protect security-sensitive cryptographic operations involving critical Wallet Unit assets.

The following sections describe the individual activation steps and the associated cryptographic assets.

1.3.2.1 Wallet Provider User account

The User must register with the Wallet Provider to create a Wallet Provider User account. The wallet provider must associate the Wallet Provider User account with the installed Wallet Instance.

The Wallet Provider associates the Wallet Unit with a User account to support Wallet Unit management, including User-requested revocation. The account does not necessarily need to contain the User's real-world identity unless required for a specific Wallet function or legal obligation.

1.3.2.2 User Authentications

The User enrols two user authentication mechanisms to the Wallet Instance. These two User authentication mechanisms are:

OS-level User authentication:

- The Wallet Instance shall require OS-level User authentication whenever the Wallet Instance is opened and before performing Wallet Unit operations. The OS-level authentication mechanism shall use possession of the User Device together with a knowledge-based or inherence-based factor.
- This local OS-based authentication shall be enforced at startup of the Wallet Instance and **before security-sensitive actions**, such as
  - generating and presenting pseudonyms,
  - accessing the transaction log via the dashboard,
  - requesting the erasure of personal data by a Relying Party;
  - other operations as specified by the applicable ARF requirements.

WSCA/WSCD-level Remote User Authentication Protocol (RUAP):

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

- This specific **Remote User Authentication Protocol (RUAP),** including the use of **User Authentication Evidence (UAE)** and **Authentication Reference Data (ARD),** is a PP-specific mechanism defined to implement the required WSCA/WSCD-level User authentication.
- The RUAP shall support **User authentication** meeting the applicable requirements for **Level of Assurance High** as specified in Commission Implementing Regulation (EU) 2015/1502. In particular, the authentication mechanism shall use at least two authentication factors from different categories and shall implement security controls such that guessing, eavesdropping, replay and manipulation of the authentication communication cannot, with high attack potential, subvert the authentication mechanism.
- The Wallet Instance must demonstrate the User Authentication Evidence (UAE), which is derived both from a **knowledge-based authentication factor** (derived from the PIN) and a **possession-based authentication factor**, which are jointly validated to achieve two-factor authentication and to prevent manipulation or forged attempts.
- Remote User Authentication Protocol (RUAP) ensure PIN Verification and ensure that the User has **sole control** over the critical assets located in the WSCD, such that these assets cannot be used without the User's involvement through the required authentication factors;
- The protocol protects the confidentiality and integrity of the authentication data during PIN verification and enrolment, and provides protection against **duplication, tampering, eavesdropping and replay attacks**, including through freshness and challenge binding;
- Each PIN verification is performed through a live and fresh interaction between the Wallet Instance and the WSCA/WSCD, preventing reuse of previously exchanged authentication information;
- During enrolment, the Wallet Instance Wallet Instance ensures that both the knowledge and possession factors are securely initialized and cryptographically bound to the **Authentication Reference Data (ARD)**. Authentication Reference Data (ARD) is derived and is shared with the WSCA /WSCD during initial enrolment. This binding preserves confidentiality and integrity of the knowledge-based authentication factor (derived from the PIN) and a possession-based authentication factor and prevents compromise also if the confidentiality of the Authentication Reference Data (ARD) is not preserved.
- A **Personal Unblocking Key (PUK)** can be integrated in the protocol to reset the knowledge-based factor in case the PIN was forgotten.
- **Per-key authentication data** shall be held by the Wallet Instance and be provided at every security-sensitive cryptographic operation of the applicable key, to ensure sole control of the Wallet User before the execution by the WSCD.
- This authentication mechanism is subsequently required whenever the Wallet Unit performs cryptographic operations involving protected assets within the WSCA/WSCD, ensuring that all such operations remain securely bound to both the user and the user device. At a minimum, in the following situations. WSCA/WSCD-level User Authentication is required:
  - Initial Generation of KAs and WIA (further detailed in section 1.3.2.3).
  - The Wallet Unit asks the User for approval of selective disclosure to a Relying Party (further detailed in section 1.3.4.2)
  - The User deletes a PID or attestation and the attested keys in their Wallet Unit (further detailed in section 1.3.5).
  - The User instructs the Wallet to request the issuance of a new PID or Attestation.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

1.3.2.3 Initial Generation of the WIA and KA

During Wallet Activation, at least one set of Wallet Unit Attestations (WUA) is issued. These attestations comprise the Key Attestation(s) (KA) and the Wallet Instance Attestation (WIA).

The KA provides information about the WSCA/WSCD and contains the cryptographic public keys of the attested key pairs. The corresponding private keys are protected by the WSCA/WSCD. The WIA provides information about the Wallet Instance and Wallet Solution, including applicable certification information. The WIA device-binding key pair may optionally be generated within the WSCA/WSCD or managed within the context of the Wallet Instance.

The WIA and KA are issued and signed by the Wallet Provider. The WSCA/WSCD shall provide cryptographic means (signing) for the Wallet Provider to authenticate the public keys before including them in the WIA or KA.

1.3.2.3.1 Key attestations (KA)

The User must perform WSCA/WSCD-level user authentication through the Wallet Instance to request the WSCA/WSCD to generate the attested keys for the Key Attestation (KA). The Key Attestation (KA) is a signed information object, generated and signed using the Wallet Provider signing key (WP_SIG_PRIV1), and contains:

- One or more attested public keys (PID/ATT_PUB_KEYs)
- Applicable certification and security information concerning the WSCA/WSCD.

The attested public keys (PID/ATT_PUB_KEYs) shall be signed by the WSCA/WSCD signing keys (WSCA/D_PRIV_KEY) allowing the Wallet Provider to authenticate the public keys before KA issuance.

Key Attestation (KA) has the following **purposes**:

- The Wallet Unit shares the KA with the PID and Attestation Providers during issuance of a device-bound PID or Attestation.
- PID Provider or an Attestation Provider verify the signature using the WP_SIG_PUB1, verified through a certificate chain, where the Provider's certificate is validated through the Access Certificate Authority (ACA) up to a trusted anchor provisioned by the relevant EUDI Wallet trust framework.
- PID Provider or an Attestation Provider will use the Applicable certification and security information of the WSCA/WSCD, to verify if the WSCA/WSCD is fit to protect the private keys for a PID or Attestation or whether it is revoked.
- PID Provider or an Attestation Provider can request a proof-of-possession, by sending a nonce via the wallet instance to the WSCA/WSCD to be signed with private keys (PID/ATT_PRIV_KEYs) corresponding to PID/ATT_PUB_KEYs. This proves that the (PID/ATT_PRIV_KEYs) corresponding to PID/ATT_PUB_KEYs are generated by and stored in the WSCA/WSCD.
- PID Provider or an Attestation Provider will include the PID/ATT_PUB_KEYs the issued PID or Attestation.

1.3.2.3.2 Wallet Instance Attestation (WIA)

The Wallet Instance Attestation (WIA) is a signed information object, signed by the Wallet Provider Key (WP_SIG_PRIV2) and contains:

- Wallet Information: Wallet Solution ID/Name, version and URIs with Certification information and additional Wallet Solution information.
- Expiration and revocation status. The technical validity period of a WIA shall be less than 24 hours.
- WIA public key: WIA_KEY_PUB.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

The WIA device-binding key pair (WIA_KEY_PRIV and WIA_KEY_PUB) may be managed either by the WSCA/WSCD or by the Wallet Instance.

The WIA device-binding public key (WIA_KEY_PUB) shall be signed by the WSCA/WSCD signing keys (WSCA/D_PRIV_KEY) allowing the Wallet Provider to authenticate the public keys before WIA issuance.

Key Attestation has the following **purposes**:

- The Wallet Unit shared Wallet Instance Attestation (WIA) with the PID and Attestation Providers.
- PID Provider or an Attestation Provider verify the signature using the WP_SIG_PUB2, verified through a certificate chain, where the Provider's certificate is validated through the Access Certificate Authority (ACA) up to a trusted anchor provisioned by the relevant EUDI Wallet trust framework.
- Sent to PID and Attestation Providers to inform them about the Wallet Instance details, such as Wallet Solution name, version, and certification of the Wallet Solution, and revocation status.
- PID Provider or an Attestation Provider will use the Wallet Identifiers to verify if the Wallet Unit belongs to a certified Wallet Solution registered in the Wallet Provider Trusted List.
- PID Provider or an Attestation Provider uses the revocation reference allowing a PID Provider or an Attestation Provider to verify if the Wallet Solution has been revoked.
- PID Provider or an Attestation Provider can request a proof-of-possession, by sending a nonce via the wallet instance to the WSCA/WSCD to be signed with private keys (WIA_KEY_PRIV) corresponding to WIA_KEY_PUB. This proves that the (WIA_KEY_PRIV) corresponding to WIA_KEY_PUB is managed by the Wallet Unit.

1.3.3 PID/Attestation Provider authentication and Issuance.

Before the PID/Attestation Provider issues the PID or the Attestation to the Wallet Unit, they will first authenticate to each other.

1.3.3.1 PID/Attestation Provider authentication to the Wallet Unit

The first step in the issuance process is the authentication of the Provider by the Wallet Instance.

During Provider authentication the Wallet Instance executes the following steps:

- Wallet Instance initiates connection with the PID/Attestation Provider.
- The Provider returns to the Wallet Instance its Access Certificate (ISS_ACC_CERT) and, where applicable, a Registration Certificate (ISS_REG_CERT).
- The Wallet Instance verifies the certificates using the applicable certificate chain and trusted anchor.
- The Wallet Instance verifies the Provider's registered identity, role and, where applicable, the credential types or services for which the Provider is registered. This information is obtained from the Registration Certificate or the applicable Registrar service.

1.3.3.2 Wallet Unit authentication to the PID/Attestation Provider

The next step in the Issuance process is for the Wallet Instance to send the PID/attestation request (OpenID4VCI -Credential Request) to the Provider, and for the Provider to authenticate the WUAs (KA & WIA).

During WIA authentication the following steps are executed:

- Wallet Instance sends the signed WIA to the Provider.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

- The Provider verifies the WIA signature using the applicable Wallet Provider public key (WP_SIG_PUB2). Trust in the Provider's public key shall be established through the applicable Trust List or LoTE.
- The Provider verifies the validity and revocation status of the WIA.
- The Provider requests proof-of-possession from the Wallet Unit of WIA_KEY_PRIV corresponding to WIA_KEY_PUB contained in the WIA. The Provider provides a nonce to the Wallet Instance. The Wallet Instance signs the nonce using WIA_KEY_PRIV, either directly or through the WSCA/WSCD where the key is stored there. The signed nonce is returned to the Provider and verified using WIA_KEY_PUB.

During KAs authentication the following steps are executed:

- Wallet Instance sends a signed KA(s) to the Provider.
- The Provider verifies the signatures using the Wallet Provider public (WP_SIG_PUB1) key. Trust in the Provider's public key shall be established through the applicable Trust List or LoTE.
- The Provider verifies the validity and revocation status of the KA.
- The Provider requires proof-of-possession from the wallet unit, of the private counterpart (PID/ATT_PRIV_KEY) of the public key in the KA (PID/ATT_PUB_KEYs), as defined in OpenID4VCI. For this, the Provider provides a nonce to the Wallet Instance. The Wallet Instance will request the WSCA/WSCD to sign this nonce with the PID/ATT_PRIV_KEY. The signed nonce is returned to the Provider and the Provider will verify the signature with PID/ATT_PUB_KEY from the KAs.
- To avoid a second request from the Wallet Instance to the WSCA/WSCD, If the Wallet Instance generates a new KA during PID/Attestation issuance the nonce can be included in the KA, and the KA is additionally signed by the PID/ATT_PRIV_KEYs matching the public keys in the KA (PID/ATT_PUB_KEYs).

The Wallet Instance sends a KA to a PID Provider or Attestation Provider only for device-bound, not for non-device-bound PIDs/Attestations.

1.3.3.3 PID/Attestation Generation

Now that the Provider has ensured that it is interacting with the Wallet Unit holding the private key corresponding to the public key embedded in the KA, has verified the applicable WSCA/WSCD or keystore security properties, and has verified the Wallet Instance through the WIA, it can generate the PID or Attestation.

For a device-bound PID or Attestation, the Provider binds the issued credential to one of the public keys contained in the applicable KA. The issued PID/Attestation is signed using the Provider's private signing key (ISS_SIG_PRIV) and is sent to the Wallet Instance using the OpenID4VCI Credential Response.

The exact cryptographic structure of the issued PID or Attestation depends on the credential format and the applicable technical specification.

The Wallet Unit supports, where applicable:

- **SD-JWT VC** for remote presentation to Relying Parties.
- **ISO/IEC 18013-5 mdoc** for proximity presentation.

Both formats support selective disclosure based on a common high-level cryptographic principle: the credential contains cryptographic commitments, typically in the form of salted hashes, that allow individual attributes to be disclosed and subsequently verified without disclosing the other attributes. The disclosed attribute and the corresponding cryptographic information allow the Relying Party to verify that the attribute was included in the credential and protected by the Provider's signature.

The detailed cryptographic structures and verification mechanisms are format-specific and are defined by the applicable SD-JWT VC and ISO/IEC 18013-5 specifications.

For device-bound credentials, the public key associated with the protected private key is included or referenced in the credential in accordance with the applicable credential format. During presentation, the Wallet Unit uses the corresponding private key to provide the required proof of possession or device binding.

1.3.4 Credential (PIDs/Attestation) Disclosure

At this point in the lifecycle, the Wallet Instance holds PIDs or Attestations and is ready to disclose selected attributes/claims to the Relying Party.

PIDs/Attestation selective disclosure includes the following steps:

- The Wallet Instance authenticates the Relying Party.
- The Wallet Instance requests User approval of the attributes/claims to be disclosed and performs selective disclosure.

1.3.4.1 Wallet Instance authenticates Relying Party

The Wallet Instance authenticates the Relying Party using an Access Certificate (RP_ACC_CERT) and obtains or verifies the applicable Relying Party registration information using a Registration Certificate (RP_REG_CERT) or the applicable Registrar's online service.

Trust in the Access Certificate and Registration Certificate is established using the applicable trusted anchors.

The Access Certificate is used to authenticate the Relying Party or intermediary participating in the presentation transaction. The Registration Certificate or Registrar service provides information about the Relying Party's registered role, Service, intended use and applicable information relevant to the presentation request.

The Wallet Instance must inform the User of the outcome of the Relying Party authentication and certificate validation.

1.3.4.2 User approval and selective disclosure

Prior to User approval, the Wallet Instance shall authenticate the User using OS-level or WSCD/WSCA-based authentication.

The Wallet Instance shall present the following information to the User and obtain explicit approval before disclosure:

- The identity of the Relying Party, and intermediary parties if present.
- Attributes/claims the Relying Party requested
- Relying Party's intended use (from Registration cert or service), and privacy policy for these attributes,
- Outcome of the evaluation of the requested attributes

The Wallet Instance shall support selective disclosure using the mechanism defined by the applicable PID/Attestation format. At a high level, the individual attributes are represented by salted cryptographic hashes that are protected by the Provider's signature. When an attribute is disclosed, the Relying Party can use the disclosed attribute and the corresponding hash information to verify that it was included in the credential, without requiring disclosure of the other attributes.

The Wallet Instance shall generate a presentation bound to the transaction, including the freshness value provided by the Relying Party, and use the WSCD/WSCA to sign the

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

presentation with the corresponding PID/Attestation private key (PID/ATT_PRIV_KEY). This provides proof-of-possession of the private key.

The Relying Party shall verify the presentation, including its authenticity, integrity, freshness and the authenticity of the Provider, using the applicable credential and presentation specifications. Trust in the Provider's public key shall be established through the applicable Trust List or LoTE.

After the transaction, the Wallet Instance shall enable the User to report unlawful or suspicious requests for personal data and to request deletion of personal data from the Relying Party.

1.3.5 Migration to a new Wallet Instance

The Wallet Unit supports secure migration of Wallet Unit contents to a new Wallet Instance. The migration process uses a protected Migration Object containing the information required to restore or re-issue applicable credentials while excluding sensitive information such as WIA, KA and private keys.

1.3.6 Revocation/Deletion

The Wallet Provider must revoke the Wallet Unit upon explicit User request, upon an explicit request from a PID Provider in accordance with the applicable revocation procedure, and when a security breach or compromise affects the trustworthiness or reliability of the Wallet Unit.

A Wallet Instance is revoked by setting the `revoked` status at all index positions specified in the WIAs issued to that Wallet Instance. A WSCD is revoked by setting the `revoked` status at all index positions specified in the corresponding KAs.

The Wallet Unit shall support secure deletion of its contents. This shall delete all attestations, logs, personal data, settings and configurations and, where connected, request the associated WSCA/WSCD and keystores to delete the cryptographic assets related to the Wallet Unit.

2 Conformance Claims

2.1 Common Criteria Conformance Claim

This PP claims conformance to the CC:2022 Revision 1 including the latest errata:

- Part1: Introduction and general model CC:2022 Revision 1
- Part2: Security functional components CC:2022 Revision 1
- Part3: Security assurance components CC:2022 Revision 1
- Part4: Framework for the specification of evaluation methods and activities CC:2022 Revision 1
- Part5: Pre-defined packages of security requirements CC:2022 Revision 1

Common Criteria conformance: CC Part 2 extended and CC Part 3 Conformant.

2.2 Package Claim

This PP claims conformance to assurance package EAL 4 augmented with AVA_VAN.5, ALC_DVS.2 and ALC_FLR.3.

2.3 Protection Profile Conformance Claim

This PP does not claim conformance to other PPs.

2.4 Conformance Statement

This PP requires strict conformance of any ST or PP that claims conformance to this PP.

2.5 Conformance claim rationale

This PP does not claim conformance to any PP. Therefore, this section is not applicable.

3 Security Problem Definition

3.1 Assets

Assets that the TOE is responsible for managing and protecting are defined in ***Table*** 1.

Table 1***—assets***

Asset	Definition	Attributes
AST.Channel_Authentication_Key_Pair	The WSCA(TOE)/WSCD shall use a Channel Authentication Key Pair, consisting of a private key securely protected by the WSCD-HSM and a corresponding public key or certificate, to authenticate the WSCA(TOE)/WSCD to the Wallet Instance. The Wallet Instance shall use its corresponding authentication key material to provide mutual authentication, while session keys established during the authentication process provide confidentiality and integrity protection.	Confidentiality (private key) Integrity Authenticity
AST.Wallet_Instance_Authentication_Data	The Wallet Instance counterpart for AST.Channel_Authentication_Key_Pair, used to establish the secure channel between the TOE and the Wallet Instance.	Integrity Authenticity
AST.Channel_Authentication_Session_ Key	Secure channel session key for the mutually authenticated secure channel established between the TOE and the Wallet Instance described in AST.Channel_Authentication_Key_Pair and AST.Wallet_Instance_Authentication_Data .	Confidentiality Integrity Authenticity
AST.Authentication_Reference_Data	Authentication Reference Data (ARD) is managed by the TOE and used to verify the User's Authentication Evidence (UAE) during the Remote User Authentication Procedure (RUAP). If the ARD is stored in an external database, the TOE shall first verify its authenticity and integrity. The TOE shall not rely solely on the database to ensure the integrity of the ARD. Confidentiality must be ensured to avoid PIN (or PUK) verification outside the TOE.	Integrity Authenticity
AST.Wallet_Provider_Authentication_Data	Authentication data used by the TOE to authenticate a Wallet Provider.	Confidentiality (if private key) Integrity Authenticity

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Asset	Definition	Attributes
AST.Attested_Key_Pairs	The attested keys pairs (PID/ATT_PRIV_KEY and PID/ATT_PUB_KEY) are generated in the WSCD-HSM and managed by the TOE. The private key is kept inside the secure boundaries of the WSCD-HSM. ▪ The private keys are never exported from the WSCD-HSM. ▪ The public parts exported and added to the Key Attestations (KA).	Confidentiality (private key) Integrity Authenticity
AST.WIA_Device_Binding_Key_Pair	WIA device-binding key pair (WIA_PRIV_KEY and WIA_PUB_KEY) are generated in the WSCD-HSM and managed by the TOE. The private key is kept inside the secure boundaries of the WSCD-HSM. ▪ The private keys are never exported from the WSCD-HSM. ▪ The public parts exported and added to the Key Attestations (KA) and the Wallet Instance Attestation (WIA). Note: The WIA device-binding key pair (WIA_KEY_PRIV and WIA_KEY_PUB) may be managed either by the WSCA/WSCD or by the Wallet Instance. WIA_KEY_PRIV is only considered an asset for the TOE in case the WSCA manages the key.	Confidentiality (private key, if managed by the TOE) Integrity Authenticity
AST.WSCA_WSCD_Signing_Key_Pair	WSCA(TOE)/WSCD signing key pair (WSCA/D_PRIV_KEY and WSCA/D_PUB_KEY) are generated in the WSCD-HSM and the private key is kept inside the secure boundaries of the WSCD-HSM. ▪ The private keys are never exported from the WSCD-HSM. ▪ The public parts are securely shared with the Wallet Provider.	Confidentiality (private key) Integrity Authenticity
AST.WSCA_BE_Software_And_Configuration_Data	The TOE software and configuration data is installed and executed on server operational environment. It is at risk from server-side compromise, insider threat and supply chain attacks. The server operational environment must enforce integrity controls.	Integrity Authenticity

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Asset	Definition	Attributes
AST.WSCD_HSM_Access_Credentials	WSCD-HSM Access Credentials are used by the TOE to authenticate to the WSCD-HSM. Compromise enables an attacker to invoke WSCD-HSM operations on behalf of the TOE. This must include per-key authorisation data, limiting the WSCD-HSM access to only a minimal set of HSM assets. Per-key authorisation data is generated during key generation, is sent to the wallet instance and afterwards provided with each operation. The per-key authorization data must be securely deleted when no longer needed by the TOE. During usage, it needs to be protected by the TOE.	Confidentiality Integrity Authenticity
AST.Wrapped_Key_Blobs	The key blobs include private keys, which confidentiality is protected by the WSCD-HSM.	Integrity Authenticity
AST.User_Data	TOE protected user data, at least including S.WalletUser identifiers, PIN (or PUK) Retry Conter and user binding information to AST.Wrapped_Key_Blobs, AST.Authentication_Reference_Data; AST.Attested_Key_Pairs, AST.WSCD_HSM_Access_Credentials, AST.WIA_Device_Binding_Key_Pair.	Integrity Authenticity
AST.Audit_Records	Cryptographically protected tamper-evident audit record for each security-relevant event.	Integrity Authenticity

3.2 Subjects

Table 2 defines the subjects that interact with the WSCA-BE.

Table 2 **— *Subjects***

Subjects	Description
S.WalletProvider	Wallet provider backed application that communicates with the TOE through a mutually authenticated channel. This subject requests operations on the TOE.
S.WalletInstance	Wallet application that communicates with the TOE over a local or external 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 action can be carried out with the users' keys. The S.WalletUser does not manage or use keys directly. All communication with the TOE is mediated by the S.WalletInstance. Each S.WalletUser has exactly one S.WalletInstance.

3.3 Threats

This section of the security problem definition shows the threats that are to be countered by the TOE. A threat consists of a threat agent, an asset and an adverse action of that threat agent on that asset.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

The threats identified against the TOE and its assets are defined in ***Table*** 3. All threats assume HIGH attack potential against the WSCA-BE, consistent with EAL4 augmented with AVA_VAN.5 and the requirements of (EU) 2024/2981.

Table 3 ***— Threats***

Treat ID	Description
T.WSCA_FW_TAMPER	An attacker who compromises the TOE — through a software vulnerability, insider threat, or supply chain attack — attempts to invoke WSCD-HSM signing operations with the AST.Attested_Key_Pairs, AST.WIA_Device_binding_Key_Pair and AST.WSCA_WSCD_ Signing_Key_Pair on behalf of a victim wallet user without that user's knowledge or consent.
T.WSCA_ATTESTATION_TAMPER	An attacker tampers with the attestation process by submitting for signing with AST.WSCA_WSCD_Signing_Key_Pair a public key that was not generated within the WSCD-HSM for the authenticated wallet user, so that the Wallet Provider includes an attacker-controlled key in a Key Attestation or Wallet Instance Attestation.
T.WSCA_DATA_TAMPER	An attacker modifies the AST.WSCA_BE_Software_And_Configuration_Data through server-side exploitation — such as privilege escalation, supply chain compromise, or insider threat — resulting in subversion of access controls, suppression of audit events or unauthorised access to AST.WSCD_HSM_Access_Credentials.
T.WSCA_NET_ATTACK	A network adversary performs active attacks (e.g. read, change, forge or replay) against the secure channel between the Wallet Instance and the WSCA-BE. This includes compromising either, AST.Channel_Authentication_Key_Pair, AST.Channel_Authentication_Session_Key or AST.Wallet_Instance_Authentication_Data used for mutually authenticating the secure channel.
T.REMOTE_AUTH_VUL	An attacker exploits a vulnerability in the Remote User Authentication protocol (RUAP)— such as a forged User Authentication Evidence (UAE), or a broken session binding — to trigger a signing operation from the WSCA-BE using AST.Attested_Key_Pairs, AST.WIA_Device_binding_Key_Pair and AST.WSCA_WSCD_ Signing_Key_Pair without the legitimate wallet user's consent. Possible RUAP attacks to be considered need to include at least guessing, eavesdropping, replay, manipulation of the authentication communication or bruteforcing the PIN (or PUK).
T.WALLET_ORACLE	An attacker captures a valid Remote User Authentication Protocol interaction — including the enrolment in which Wallet Instance shares the AST.Authentication_Reference_Data Authentication Reference Data (ARD) with the WSCA-BE(TOE) — and brute forces the PIN by using the Wallet Instance to generate indefinite User Authentication Evidence (UAE) and verify it against the AST.Authentication_Reference_Data, without involvement of the WSCA-BE(TOE).
T.WSCD_NET_ATTACK	A network adversary performs active attacks (e.g. read, change, forge or reply) against the secure channel between the WSCD and the WSCA-BE. This includes compromising AST.WSCD_HSM_Access_Credentials.
T.MULTI_TENANT	An attacker manages to circumvent TOE user isolation mechanisms to gain unauthorized access to another users assets, using their AST.Attested_Key_Pairs and AST.WIA_Device_binding_Key_Pair.
T.KEY_EXTRACT	An attacker manages to abuse the WSCD-HSM client interface, using the AST.WSCD_HSM_Access_Credentials to extract HSM-stored private keys (e.g. AST.Attested_Key_Pairs, AST.WIA_Device_binding_Key_Pair and AST.WSCA_WSCD_ Signing_Key_Pair) from the WSCD_HSM.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Treat ID	Description
T.DERIVE	An attacker manages to derive any private key (AST.Attested_Key_Pairs, AST.WIA_Device_binding_Key_Pair, AST.WSCA_WSCD_Signing_Key_Pair), e.g. from key specific information like the public key or AST.Wrapped_Key_Blobs.
T.WSCA_USER_DATA_TAMPER	An attacker manages to tamper with the S.WalletUser's or S.WalletInstance's AST.User_Data or AST.Wrapped_Key_Blobs to gain access to another S.WalletUser's private keys.
T.PROVIDER_IMPERSONATION	An attacker forges or steals AST.Wallet_Provider_Authentication_Data in order to impersonate a Wallet Provider and get access to Wallet Provider specific functionality.
T.PROVIDER_NET_ATTACK	An attacker forges or steals AST.Wallet_Provider_Authentication_Data in order to impersonate a Wallet Provider by attacking the mutual authenticated secure channel between the Wallet Provider and the WSCA.
T.SIDE_CHANNEL	A co-tenant on shared infrastructure components exploits side channels in the WSCA-BE's processing of requests and responses to infer sensitive information about wallet user credentials.
T.AUDIT_TAMPER	An attacker suppresses, modifies, deletes or causes incorrect attribution of security-relevant audit records, preventing detection or investigation of unauthorized WSCA-BE or WSCD-HSM operations.

3.4 Organisational Security Policies

The organisational security policies (OSPs) as defined ***Table*** 4, shall apply to the WSCA-BE.

Table 4***— Organisational Security Policies***

ID	Description
OSP.CRYPTO_STANDARDS	The WSCA-BE shall use only cryptographic algorithms as mandated by the applicable implementing acts or as approved under the ECCG Agreed Cryptographic Mechanisms (ACM) or equivalent successor documents, meeting minimum key length and algorithm selection requirements. Where a cryptographic mechanism required by the TOE is **not yet included in the ECCG Agreed Cryptographic Mechanisms**, its use shall be subject to an **appropriate security assessment and documented justification** demonstrating that the mechanism provides sufficient security for the intended purpose and security strength. Such a mechanism shall only be used where its security has been established in accordance with the **applicable certification or evaluation framework**, and shall be replaced or updated when an applicable ECCG mechanism becomes available.
OSP.VULN_MGMT	The TOE developer shall operate a vulnerability management process compliant with EN ISO/IEC 30111:2019 and the requirements of Article 5 of (EU) 2024/2981 or any of their successor documents.
OSP.INFRA_SEC	The server infrastructure hosting the WSCA-BE shall be operated under established policies and documented procedures that include physical access controls, privileged access management and change management processes appropriate for managing credentials at the eIDAS level of assurance high. Provider's conformity with these requirements are in-scope of the national certification scheme under TR 5.
OSP.WSCD_CERT	The WSCD-HSM shall be certified against an appropriate HSM Common Criteria Protection Profile. An appropriate Protection Profile is the CEN EN PP 419 221-5 or a different applicable Protection Profile providing at least equivalent functionality and assurance. Private keys stored in the HSM must be protected at least following the "assigned key" definition of the CEN EN PP 419 221-5.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

ID	Description
OSP.KEY_ISOLATION	Each S.WalletUsers's private keys shall be secured within the secure boundaries of the WSCD-HSM and bound to the S.WalletUsers's identity in the WSCA-BE's credential store. Key misuse shall not be possible without compromising the AST.WSCD_HSM_Access_Credentials or the WSCD-HSM itself.
OSP.AUDIT	The TOE shall maintain a cryptographically protected tamper-evident (non-repudiable) AST.Audit_Records of all security-relevant operations attributes to individual wallet user and transactions. The audit records shall be cryptographically integrity and authenticity protected. For each S.WalletInstance requested operation, the TOE shall generate evidence of origin and receipt that can be verified by a S.WalletProvider.
OSP.TOE_AUTHENTICATION	The TOE shall be able to uniquely identify itself, its version and its configuration. Additionally, the TOE shall be able to authenticate itself to the user.
OSP.SECURE_ERASURE	Critical assets held by the WSCA-BE shall be securely erased on revocation, on deprovisioning, factory reset, and on an authorised deletion request.
OSP.MIGRATION	An authenticated Wallet User shall be allowed to migrate to a new wallet instance.

3.5 Assumptions

The assumptions, as defined in ***Table*** 5, describe the security properties of the operational environment. Security Targets shall provide evaluation evidence to confirm these assumptions are satisfied in the specific deployment.

Table 5— ***Assumptions***

ID	Description
A.UAE_and_ARD	It is assumed that the Wallet Instance cryptographically derives the User Authentication Evidence (UAE) from a knowledge-based factor (derived from the PIN) and a possession-based factor. A PUK is used to reset the knowledge-based factor in case the PIN was forgotten. It is assumed that User Authentication Evidence (UAE) and the Authentication Reference Data (ARD) preserve confidentiality and integrity of the knowledge-based factor (derived from the PIN) and a possession-based factor. Where part of the Wallet User authentication is verified outside the TOE, the external verification meets the requirements of CIR (EU) 2015/1502 Annex, sections 2.2.1 and 3.1, at level high. In particular the external mechanism provides dynamic authentication, protects against duplication and tampering by an attacker with high attack potential, and secures stored verification data against offline analysis. It is assumed that the Wallet Instance generate and uses the required cryptographic keys within the secure keystore. Additionally, the Wallet Instance app is assumed to be from an official app store and/or authenticated in a secure way. Additionally, the wallet shall only run on secure devices (including root/emulator detection, secure key storage capabilities, OS Version without related vulnerabilities) The wallet instance shall be certified according to (EU) 2024/2981.
A.PHYSICAL_PROTECTION	The TOE is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security. This protection is assumed to be sufficient to protect the TOE and the data it contains. As a result, the PP does not include any requirements on physical tamper protection or other physical attack mitigations (such as power analysis or EM analysis). The PP does not expect the product to defend against physical access to the TOE that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

ID	Description
A.TRUSTED_ADMIN	The Administrators are assumed to be trusted and to act in the best interest of security for the organization, following best practices for high trust environments. This includes appropriately trained, following policy, follow secure processes, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device.
A.TRUSTED_WALLET_ USER	It is assumed that the S.WalletUser will not disclose their proof of knowledge.
A.TRUSTED_WALLET_ PROVIDER	S.WalletProvider is assumed to be properly trained and trusted and will not disclose their AST.Wallet_Provider_Authentication_Data. S.WalletProvider is assumed to restrict admissible AST.Authentication_Reference_Data such that S.WalletUser can reliably protect their user authentication credentials.
A.AUDIT_LOG_SERVER	Audit log server is located in the TOE environment which is able to receive audit log messages from the TOE. The environment shall ensure that log data is not deleted.
A.DATABASE	External databases used for storage of TOE assets shall be located in the secure environment.
A.GATEWAY	Internet facing gateways are excluded from the TOE scope and shall be used for, at least, DDoS protection, intrusion detection, antivirus functionality and firewall functionality.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

4 Security Objectives

4.1 Security Objectives for the TOE

The security objectives for the TOE are set out in ***Table*** 6.

Table 6 ***— TOE security objectives***

ID	Statement
OT.KEY_PROTECT	The WSCA-BE shall ensure that all private keys associated with wallet users are generated within and never exported from the secure boundaries of the WSCD-HSM. Only cryptographic algorithms as mandated by the applicable implementing acts or as approved under the ECCG Agreed Cryptographic Mechanisms (ACM) or equivalent successor documents, meeting minimum key length and algorithm selection requirements shall be used. The TOE itself shall only store a minimum set private and symmetric keys if necessary for basic TOE functionality, namely session keys and secure channel authentication data. The WSCA-BE shall hold only opaque key handles and shall enforce that all signing and key agreement operations are executed by the WSCD-HSM. The WSCA-BE shall protect at least AST.Authentication_Reference_Data, AST.Wallet_Instance_Authentication_Data and AST.Wallet_Provider_Authentication_Data.
OT.KEY_ISOLATION	The WSCA-BE shall enforce per-wallet-unit key isolation by associating each key handle with the authenticated wallet user identity before invoking any WSCD-HSM operation. The WSCA-BE shall use the WSCA/WSCD signing key, only for signing the public key of key pairs generated within the WSCD-HSM and bind it to the wallet user. The WSCA-BE uses and protects per-key WSCD-HSM authorization credentials that only grant access to user keys connected to this credential. Per-key authorisation data is generated by the WSCD-HSM during key generation, then sent to the wallet instance and afterwards provided with each operation. The per-key authorization data is considered a TOE asset must be protected by the TOE during usage, and securely deleted when no longer needed by the TOE.
OT.SECURE_DELETION	Any TOE-stored assets shall be deleted in a secure manner. This includes a mechanism for secure factory reset.
OT.USER_AUTH	The WSCA-BE shall verify, before invoking any WSCD-HSM operation, that the request is accompanied by valid User Authentication Evidence (UAE) that is: (a) cryptographically bound to the specific wallet user's registered Authentication Reference Data (ARD); (b) bound to the specific transaction context; (c) within its validity window; and (d) not previously used. (e) resilience against tampering or brute force attacks on the PIN (or PUK) verification. The WSCA-BE shall reject requests lacking valid evidence and record each rejection in the audit log.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

ID	Statement
OT.PROVIDER_AUTH	The TOE shall mandate authentication of the S.WalletProvider before any provider related operation can be invoked.
OT.API_CONTROL	The WSCA-BE shall ensure that authenticated S.WalletUser can only invoke operations on their own resources, unauthenticated requests are rejected before any sensitive processing. Each operation using keys must include the specific per-key authorization data to be provided by the S.WalletInstance.
OT.WSCA_SECURE_CHANNEL	The WSCA-BE(TOE) shall enforce a mutually authenticated secure channel between the TOE and the Wallet Instance. The WSCA-BE and the Wallets Instances will route all communication through the secure channel. The WSCA-BE(TOE) shall bind each transaction identifier to the AST.Wallet_Instance_Authentication_Data, preventing use of stolen User Authentication Evidence from a different client.
OT.WSCD_SECURE_CHANNEL	The WSCA-BE(TOE) shall enforce a mutually authenticated secure channel between the TOE and the WSCD. The WSCA-BE and the WSCD will route all communication through the secure channel.
OT.WALLET_PROVIDER_SECURE_CHANNEL	The WSCA-BE(TOE) shall enforce a mutually authenticated secure channel between the TOE and the Wallet Provider. The WSCA-BE and the Wallet Provider will route all communication through the secure channel.
OT.MULTI_TENANT	The WSCA-BE shall implement strict isolation between wallet users such that an authenticated request from Wallet User A cannot access credentials, invoke WSCD-HSM operations using keys belonging to, or read records of Wallet User B.
OT.AUDIT	The WSCA-BE shall generate a cryptographically protected tamper-evident audit record for each security-relevant event, including wallet user identifier; HSM key handle; relying party identifier (where applicable); User Authentication Evidence reference; and outcome and potential threats. Audit records shall be cryptographically integrity and authenticity protected. For each Wallet Instance requested operation, the TOE shall generate evidence of origin and receipt of the request so that can be verified by a Wallet Provider. Wallet Provider actions shall be logged.
OT.SELF_PROT	The WSCA-BE shall detect and refuse to start if its binary or configuration has been modified since the last integrity baseline. WSCD-HSM Access Credentials shall not be stored in plaintext within the WSCA-BE process or on storage.
OT.PROTECTED_STORAGE	The TOE shall protect the integrity and authenticity of all user data stored on the WSCA or its operative environment. This includes the key blobs, meta data, all user bindings and all authentication data stored outside of the WSCD. Additionally, all session keys and non-WSCD stored private keys shall be confidentiality protected. Any sensitive temporary data, including session keys and per-key authorization data, needs to be securely delete if not longer used.
OT.AUTHENTICATION	The TOE is able to uniquely identify itself, its version and its configuration. Additionally, the TOE shall be able to authenticate itself.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

4.2 Security Objectives for the Operational Environment

The security objectives for the operational environment are set out in ***Table*** 7.

***Table** 7 — Operational environment security objectives*

ID	Statement
OE.UAE_and_ARD	The Wallet Instance will cryptographically bind the User Authentication Evidence (UAE) from a knowledge-based factor (derived from the PIN) and a possession-based factor. A PUK is used to reset the knowledge-based factor in case the PIN was forgotten. The Remote User Authentication Protocol (RUAP) implemented between User Authentication Evidence (UAE) and the Authentication Reference Data (ARD) will preserve confidentiality and integrity of the knowledge-based factor (derived from the PIN) and a possession-based factor (also when these are disclosed). Where part of the Wallet User authentication is verified outside the TOE, the external verification shall meet the requirements of CIR (EU) 2015/1502 Annex, sections 2.2.1 and 3.1, at level high. In particular the external mechanism provides dynamic authentication, protects against duplication and tampering by an attacker with high attack potential, and secures stored verification data against offline analysis. The Wallet Instance will generate and uses the required cryptographic keys within the secure keystore of the user device.
OE.RESISTANT_ARD	The Wallet Provider restricts admissible ARD such that Wallet User can reliably protect their authentication credentials. ARD shall be constructed in way that knowledge-based factor cannot be verified without a secret held by the WSCA-BE or the WSCD.
OE.HSM_CERT	The operational environment shall provide a WSCD-HSM certified against an appropriate HSM Common Criteria Protection Profile. An appropriate Protection Profile is the CEN EN PP 419 221-5 or a different applicable Protection Profile providing at least equivalent functionality and assurance. Private keys stored in the HSM must be protected at least following the 'assigned key' definition of the CEN EN PP 419 221-5.
OE.INFRA_SEC	The operational environment shall include physical and logical security controls for the WSCA-BE(TOE) infrastructure meeting OSP.INFRA_SEC, including privileged access management enforcing separation of duties, intrusion detection and change management. External databases (used for asset storage), gateways (used for at least DDoS protection) and the audit log server (OE.AUDIT_LOG_SERVER) shall be part of this secure operational environment. The operational environment needs to provide a reliable time source. The WSCA-BE(TOE) shall be operated on a hardened server using a least-privilege execution model.
OE.ISOLATION	The operational environment shall ensure that the WSCA-BE (TOE) process is isolated from other workloads, preventing memory access, network interception or storage tampering by co-hosted services.
OE.PHYSICAL_PROTECTION	The TOE is physically protected in its operational environment and not subject to physical attacks that compromise the security. The TOE underlying hardware is be hardened against remote side-channel attacks and microarchitectural attacks.
OE.VULNERABILITY_MANAGEMENT	The WSCA-BE developer operates a vulnerability management process compliant with EN ISO/IEC 30111:2019 or its successors and the requirements of Article 5 of (EU) 2024/2981.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

ID	Statement
OE.GENUINE_WALLET_INSTANCE	The Wallet Instance app is sourced to be from an official app store and/or authenticated in a secure way. Additionally, the wallet shall only run on secure devices (including root/emulator detection, secure key storage capabilities, OS Version without related vulnerabilities) The wallet instance shall be certified according to (EU) 2024/2981. Each Wallet user has exactly one Wallet Instance.
OE.SECURE_UPDATE	The wallet provider shall ensure that WSCA-BE(TOE) software updates are cryptographically signed, that the deployment pipeline verifies signatures before installation and that a SBOM is maintained for each deployed version.
OE.TRUSTED_ADMIN	The Administrators are trusted and to act in the best interest of security for the organization, following best practices for high trust environments. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The TOE is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security.
OE.TRUSTED_WALLET_USER	The Wallet User will not share or disclose their knowledge or possession authentication factors with 3rd parties.
OE.TRUSTED_WALLET_PROVIDER	All Wallet Provider personnel shall be properly trained and trusted, and shall not disclose the TOE-related authentication data. The Wallet Provider shall protect the public part of AST.WSCA_WSCD_Signing_Key_Pair, and the complete signing key chain used to issue the Key Attestation and the Wallet Instance Attestation, against unauthorised disclosure, modification and substitution. The Wallet Provider shall verify the authenticity of attested public keys received from the WSCA-BE before including them in a Key Attestation or a Wallet Instance Attestation. The Wallet Provider shall preserve the confidentiality and integrity of the Key Attestation and the Wallet Instance Attestation throughout the attestation process. The Wallet Provider shall ensure that any process by which a Wallet User assets are migrated, it requires identity re-verification at an assurance level equivalent to the assurance level of the Wallet Unit before the reset or migration is permitted.
OE.AUDIT_LOG_SERVER	Audit log server is located in the secure TOE environment as described in OE.INFRA_SEC which is able to receive audit log messages from the TOE. The environment shall ensure that log data is not deleted.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

5 Extended components definition

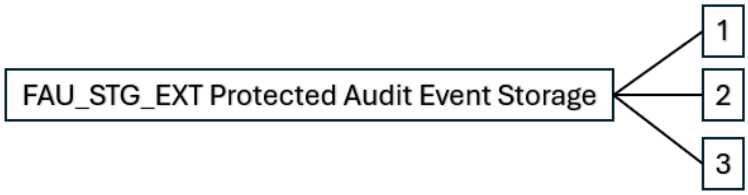
5.1 Protected Audit Event Storage (FAU_STG_EXT)

Family Behaviour

This component defines the requirements for the TSF to be able to securely transmit audit data between the TOE and an external IT entity. This is a new family defined for the FAU class.

Note: *FAU_STG_EXT is inspired by* 'collaborative Protection Profile for Network Devices', version 4.0, 2025-11-25.

Component Levelling



FAU_STG_EXT.1	Protected audit event storage requires the TSF to use a trusted channel implementing a secure protocol.
FAU_STG_EXT.2	Counting lost audit data requires the TSF to provide information about audit records affected when the audit log becomes full.
FAU_STG_EXT.3	Notification in Case of Possible Audit Data Loss requires the TSF to generate a warning before the audit trail exceeds the local storage capacity.

Management: FAU_STG_EXT.1, FAU_STG_EXT.2, FAU_STG_EXT.3

The following actions could be considered for the management functions in FMT:

a) There are no management functions foreseen.

Audit: FAU_STG_EXT.1, FAU_STG_EXT.2, FAU_STG_EXT.3

The following actions should be auditable for FAU_GEN Security audit data generation and if FAU_GEN_EXT is included in the PP/ST:

a) There are no auditable events foreseen.

5.1.1 FAU_STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1	Protected Audit Event Storage
Hierarchical to:	No other components.
Dependencies:	FAU_GEN.1 Audit data generation FAU_COP.1 Cryptographic operation
FAU_STG_EXT.1.1	The TSF shall be able to transmit the generated audit data to an external IT entity.
FAU_STG_EXT.1.2	The TSF shall be able to store generated audit data on the TOE itself.
FAU_STG_EXT.1.3	The TSF shall maintain a [selection: log file, database, buffer, [assignment: other local logging method]] of audit records in the event that an interruption of communication with the remote audit server occurs.

FAU_STG_EXT.1.4	The TSF shall be able to store [selection: persistent, non-persistent] audit records locally with a minimum storage size of [assignment: number of records and/or file/buffer size(s)].
FAU_STG_EXT.1.5	The TSF shall reject remote API requests from Wallet Instances when the local storage space for audit data is full.
FAU_STG_EXT.1.6	The TSF shall ensure the authenticity and integrity of the audit data when stored on an external IT entity or the TOE itself with cryptographic means.

5.1.2 FAU_STG_EXT.2 Counting Lost Audit Data

FAU_STG_EXT.2	Counting Lost Audit Data
Hierarchical to:	No other components.
Dependencies:	FAU_GEN.1 Audit data generation FAU_STG_EXT.1 Protected Audit Event Storage
FAU_STG_EXT.2.1	The TSF shall provide information about the number of [selection: dropped, overwritten, [assignment: other information]] audit records the case where the local storage has been filled and the TSF takes one of the actions defined in FAU_STG_EXT.1.5.

5.1.3 FAU_STG_EXT.3 Action in Case of Possible Audit Data Loss

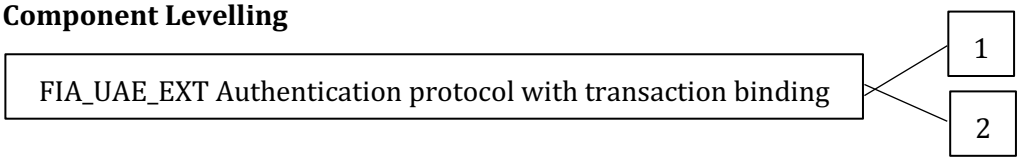
FAU_STG_EXT.3	Action in Case of Possible Audit Data Loss
Hierarchical to:	No other components.
Dependencies:	FAU_GEN.1 Audit data generation FAU_STG_EXT.1 Protected Audit Event Storage
FAU_STG_EXT.3.1	The TSF shall generate a warning to inform the [assignment: user to be informed] before the audit trail exceeds the local audit trail storage capacity.

5.2 Authentication protocol with transaction binding FIA_UAE_EXT

Family Behaviour

The following extended SFR family FIA_UAE_EXT is defined to address the specific security requirement for remote, transaction-bound user authentication evidence. This is a new family defined for the FIA class.

Component Levelling



FIA_UAE_EXT.1	Requires the TSF to issue a challenge, bind it to a transaction context, and verify returned evidence against the user's authentication reference data before performing a controlled operation.
FIA_UAE_EXT.2	Records issued challenges and accepted evidence, and to deny operations where the evidence relies on an unknown, already consumed or expired challenge.

Management: FIA_UAE_EXT.1, FIA_UAE_EXT.2

The following actions could be considered for the management functions in FMT:

a) There are no management functions foreseen.

Audit: FIA_UAE_EXT.1, FIA_UAE_EXT.2

The following actions should be auditable for FAU_GEN Security audit data generation:

a) Minimal: Failure of verification of authentication evidence
b) Basic: success and failure of verification of authentication evidence, and the reason for failure

5.2.1 FIA_UAE_EXT.1 Transaction-bound authentication evidence

FIA_UAE _EXT.1	Transaction-bound authentication evidence
Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification FIA_ATD.1 User attribute definition [FCS_RBG.1 Random bit generation or FCS_RNG.1 Generation of random] FCS_COP.1 Cryptographic operation FPT_STM.1 Reliable time stamps
FIA_UAE_EXT.1.1	The TSF shall generate, for each request to perform an operation in [assignment: list of operations requiring authentication evidence], a challenge with a minimum of [assignment: number of bits] of entropy obtained from a random bit generation function meeting [selection: FCS_RBG.1, FCS_RNG.1].
FIA_UAE_EXT.1.2	The TSF shall construct, for each challenge, a transaction context comprising the challenge, the identity of the user, the requested operation, and [assignment: other transaction attributes], and shall associate that transaction context with the communication session on which the request was received.
FIA_UAE_EXT.1.3	The TSF shall require that authentication evidence cryptographically bound to the transaction context of the requested operation be presented before the TSF performs any operation in [assignment: list of operations requiring authentication evidence].
FIA_UAE_EXT.1.4	The TSF shall verify presented authentication evidence against the authentication reference data associated with the identified user and shall verify that the evidence demonstrates that each authentication factor in [assignment: list of authentication factors] was verified.
FIA_UAE_EXT.1.5	The TSF shall verify that presented authentication evidence was received no more than [assignment: validity period] after the corresponding challenge was issued.
FIA_UAE_EXT.1.6	The TSF shall deny the requested operation if any verification required by FIA_UAE_EXT.1.3, FIA_UAE_EXT.1.4 or FIA_UAE_EXT.1.5 fails.

5.2.2 FIA_UAE_EXT.2 Single-use enforcement for authentication evidence

FIA_UAE _EXT.2	Single-use enforcement for authentication evidence
Hierarchical to:	No other components.
Dependencies:	FIA_UAE_EXT.1 Transaction-bound authentication evidence FPT_STM.1 Reliable time stamps
FIA_UAE_EXT.2.1	The TSF shall maintain a nonce registry covering all transaction identifiers issued and all evidence received within the validity window defined in FIA_UAE_EXT.1.5.
FIA_UAE_EXT.2.2	The TSF shall deny the requested operation if the presented authentication evidence refers to a challenge that was not issued by the TSF, that has already been used in successful verification, or whose validity period has expired.

6 Security Functional Requirements

All SFRs are drawn from CC Part 2 unless identified as extended components. Operations are marked as following:

- Assignment (A): indicated with text in *italic*. If assignments are possible options for the ST author, it is marked as [assignment: XYZ];
- Refinement (R): indicated with text in **bold**, and ~~strikethrough;~~
- Selection (S): selection values adopted in the PP is underlined. If selections are possible options for the ST author, it is marked as [selection: XYZ];
- Iteration (I): indicated by adding a string starting with '/' (e.g. 'FCS_COP.1/Hash')

6.1 Class FAU — Security Audit

6.1.1 FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1	The TSF shall generate an audit record of the following auditable events: a) TOE start-up and shutdown; b) All auditable events for the not specified level of audit; *c) each signing operation, including the wallet user identifier;* *d) authentication failure;* *e) wallet instance and user lifecycle events (registration, suspension, deprovisioning);* *f) administrative and management operations (FMT_SMF.1); and* *g) all self-test results (FPT_TST.1) and secure initialization (FPT_INI.1).* *h) failures listed in FPT_FLS.1 and FPT_RCV.4.* *i) warning that audit trail is near full capacity (FAU_STG_EXT.3)* *j) failing checks related to transaction-bound authentication evidence (FIA_UAE_EXT.1)* *k) failure of establishing or detected attacks on a secure or trusted channel (FTP_PRO.3/* and FPT_ITC.1/*)* *l) failure of identification or authentication.* *m)detection of integrity or authenticity errors in stored user data.* *n) migration of user key bindings from one wallet user to another by the Wallet Provider* *o) [assignment: other specifically defined auditable events].*
FAU_GEN.1.2	The TSF shall record within the audit data at least the following information: a) Date and time of the auditable event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; b) For each auditable event type, based on the auditable event definitions of the functional components included in the PP, PP-Module, functional package or ST, [assignment: other audit relevant information].

6.1.2 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.

6.1.3 FAU_SAR.1 Audit review

FAU_SAR.1.1	The TSF shall provide *Wallet Provider* with the capability to read [assignment: list of audit information] from the audit data.
FAU_SAR.1.2	The TSF shall provide the audit data in a manner suitable for the user to interpret the information.

6.1.4 FAU_STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1.1	The TSF shall be able to transmit the generated audit data to an external IT entity.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

FAU_STG_EXT.1.2	The TSF shall be able to store generated audit data on the TOE itself.
FAU_STG_EXT.1.3	The TSF shall maintain a [selection: log file, database, buffer, [assignment: other local logging method]] of audit records in the event that an interruption of communication with the remote audit server occurs.
FAU_STG_EXT.1.4	The TSF shall be able to store [selection: persistent, non-persistent] audit records locally with a minimum storage size of [assignment: number of records and/or file/buffer size(s)].
FAU_STG_EXT.1.5	The TSF shall reject remote API requests from Wallet Instances when the local storage space for audit data is full.
FAU_STG_EXT.1.6	The TSF shall ensure the authenticity and integrity of the audit data when stored on an external IT entity or the TOE itself with cryptographic means.

6.1.5 FAU_STG_EXT.2 Counting Lost Audit Data

FAU_STG_EXT.2.1	The TSF shall provide information about the number of dropped audit records the case where the local storage has been filled and the TSF takes one of the actions defined in FAU_STG_EXT.1.5.

6.1.6 FAU_STG_EXT.3 Notification in Case of Possible Audit Data Loss

FAU_STG_EXT.3.1	The TSF shall generate a warning to inform *[selection: Wallet Provider, Technical Administrator]* before the audit trail exceeds the local audit trail storage capacity.

Application Note 1

The technical administrator is not a user role for the TOE itself. This administrator is responsible for the maintenance of the operative environment, including the underlying server

6.2 Class FCO — Communication

6.2.1 FCO_NRO.2 Enforced proof of origin

FCO_NRO.2.1	The TSF shall enforce the generation of evidence of origin for transmitted *API requests from the Wallet Instance handled by the TOE* at all times.
FCO_NRO.2.2	The TSF shall be able to relate the *Wallet Instance identifier* of the originator of the information, and the *transaction identifier, UTC timestamp* of the information to which the evidence applies.
FCO_NRO.2.3	The TSF shall provide a capability to verify the evidence of origin of information to *the Wallet Provider* given *the corresponding audit record is retained by the audit log server*.

Application Note 2

The TSF shall enforce generation of evidence of origin for all signing operations. The evidence shall be a digital signature computed by the WSCD-HSM over the hash submitted by the Wallet Instance. The audit record shall bind the wallet instance identifier, the transaction identifier, and the UTC timestamp to the signing event, providing non-repudiation of the signing operation.

6.2.2 FCO_NRR.2 Enforced proof of receipt

FCO_NRR.2.1	The TSF shall enforce the generation of evidence of receipt for received API *requests from the Wallet Instance handled by the TOE* at all times.
FCO_NRR.2.2	The TSF shall be able to relate the *session identifier* of the recipient of the information, and the *Wallet Instance's certificate, public key or thumbprint and the transaction identifier* of the information to which the evidence applies.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

FCO_NRR.2.3	The TSF shall provide a capability to verify the evidence of receipt of information to *the Wallet Provider* given *the corresponding audit record is retained by the audit log server*.

Application Note 3

The TSF shall record evidence of receipt of each signing request, including the session identifier, the Wallet Instance's certificate, public key or thumbprint and the transaction identifier.

6.3 Class FCS — Cryptographic Support

6.3.1 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 4

The ST author shall model all required key generation mechanisms implemented by the TOE itself. Mechanisms implemented by the WSCD are not in scope of this SFR and the WSCD ST shall be referenced and this SFR does not have to be claimed. All cryptographic mechanisms shall follow OSP.CRYPTO_STANDARDS.

6.3.2 FCS_CKM.2 Cryptographic key distribution

FCS_CKM.2.1	The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: cryptographic key distribution method] that meets the following: [assignment: list of standards].

Application Note 5

The ST author shall model all required key distribution mechanisms implemented by the TOE itself. Mechanisms implemented by the WSCD are not in scope of this SFR and the WSCD ST shall be referenced and this SFR does not have to be claimed. All cryptographic mechanisms shall follow OSP.CRYPTO_STANDARDS.

6.3.3 FCS_CKM.5 Cryptographic key derivation

FCS_CKM.5.1	The TSF shall derive cryptographic keys [assignment: key type] from [assignment: input parameters] in accordance with a specified key derivation algorithm [assignment: key derivation algorithm] and specified cryptographic key sizes [assignment: list of key sizes] that meet the following: [assignment: list of standards].

Application Note 6

The ST author shall model all required key derivation mechanisms implemented by the TOE itself. Mechanisms implemented by the WSCD are not in scope of this SFR and the WSCD ST shall be referenced and this SFR does not have to be claimed All cryptographic mechanisms shall follow OSP.CRYPTO_STANDARDS.

6.3.4 FCS_CKM.6 Timing and event of cryptographic key destruction

FCS_CKM.6.1	The TSF shall destroy *all TSF generated and/or maintained keys and key material,* ***including per-key authorisation data*** when no longer needed, *during revocation, deprovisioning, factory reset or authorized deletion is requested*.
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].

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

6.3.5 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 [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

Application Note 7

The ST author shall model all required cryptographic mechanisms operations by the TOE itself. Operations implemented by the WSCD are not in scope of this SFR and the WSCD ST shall be referenced. All cryptographic mechanisms shall follow OSP.CRYPTO_STANDARDS.

6.3.6 FCS_RBG.1 Random bit generation (RBG)

FCS_RBG.1.1	The TSF shall perform deterministic random bit generation services using [assignment: RBG algorithm] in accordance with [assignment: list of standards] after initialization with a seed.
FCS_RBG.1.2	The TSF shall use a [selection: TSF noise source [assignment: name of noise source], TSF interface for seeding] for initialized seeding.
FCS_RBG.1.3	The TSF shall update the RBG state by [selection: reseeding, uninstantiating and re instantiating] using a [selection: TSF noise source [assignment: name of noise source], TSF interface for seeding] in the following situations: [selection: ~~— never;~~ ~~— on demand;~~ — on the condition: [assignment: condition]; — after [assignment: time]] in accordance with [assignment: list of standards].

Application Note 8

The TSF shall either use the WSCD-HSM's certified DRBG or the server OS's CSPRNG (seeded from hardware entropy) for generation of random numbers. In case the HSM DRBG is used, the HSM certification shall be referenced. Otherwise, this SFR needs to be claimed by the ST author (in this case the ST author shall additionally add FCS_DRB.2 or FCS_DRB.3, depending in the implementation of the actual TOE). Additionally, the ST author shall cover the DRBG in FPT_TST.1 and FPT_FLS.1.

6.4 Class FDP — User Data Protection

6.4.1 FDP_ACC.1/User_Creation Access Control Policy

FDP_ACC.1.1/User_ Creation	The TSF shall enforce the *User Creation SFP* on *- Subject: Wallet Provider or Wallet Instance* *- Objects: Wallet User* *- Operations: Create new user* *The TOE creates a Wallet User with information provided by the subject.*

6.4.2 FDP_ACF.1/User_Creation Access control functions

FDP_ACF.1.1/User_ Creation	The TSF shall enforce the *User Creation SFP* to objects based on the following: *whether the subject is a Wallet Provider or Wallet Instance authorised to create a new Wallet User*.
FDP_ACF.1.2/User_ Creation	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: o*nly a Wallet Provider or Wallet Instance who has been authorised for creation of new users can carry out the operation*.
FDP_ACF.1.3/User_ Creation	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *none.*

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

FDP_ACF.1.4/User_ Creation	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects].

6.4.3 FDP_ACC.1/Key_Management Access Control Policy

FDP_ACC.1.1/Key_ Creation	The TSF shall enforce the *Key Creation SFP* on *- Subject: Wallet Instance* *- Objects: Asymmetric key pairs or symmetric keys* *- Operations: Create or delete new key or key pair* *Wallet Instances instructs the TOE to request the WSCD to generate/delete new keys/key pairs and assigns it to a Wallet User.*

6.4.4 FDP_ACF.1/Key_Management Access control functions

FDP_ACF.1.1/Key_ Creation	The TSF shall enforce the *Key Creation SFP* to objects based on the following: *the wallet instance is authorised to create, delete a symmetric key or asymmetric key pairs for a Wallet User.*
FDP_ACF.1.2/Key_ Creation	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *the wallet instance requesting the creation of the key or key pair is authorised to carry out the operation.*
FDP_ACF.1.3/Key_ Creation	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *none*.
FDP_ACF.1.4/Key_ Creation	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects].

6.4.5 FDP_ACC.1/Operation Access Control Policy

FDP_ACC.1.1/Operation	The TSF shall enforce the *Operation SFP* on *- Subject: Wallet User* *- Objects: Wallet User user-bound assets (including keys or key pairs)* *- Operations: (Per-key-authorized) Operations with Wallet User user-bound assets* *The TOE instructs the WSCD to carry out operations on and with the Wallet Users assets only where it corresponds to an assets within the WSCD-HSM that is bound to the identified Wallet User using per-key-authorization data provided by the wallet instance.* *- Subject: Wallet User* *- Objects: Wallet User user-bound assets (including keys or key pairs)* *- Operations: PIN reset* *The TOE instructs the WSCD to carry out PIN reset when the correct PUK was provided.* *- Subject: Wallet User* *- Objects: Wallet User user bound assets* *- Operation: Wallet User can clear or delete the Wallet User bound assets or migrate them to a different Wallet Instance. (This does not include the WSCD-stored plaintext private keys, only their reference, keyblobs and management).*

6.4.6 FDP_ACF.1/Operation Access control functions

FDP_ACF.1.1/Operation	The TSF shall enforce the *Operation SFP* to objects based on the following: *whether the subject is a Wallet User or Wallet Provider authorised to carry out the operation or resets the PIN using the correct PUK.*

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

FDP_ACF.1.2/Operation	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: • *only an authorized Wallet User is allowed to carry out operations using the keys or key pairs* • *a Wallet User is only allowed to carry out operations using their user-bound keys.* • *the authentication evidence is valid and bound to the current transaction.* • *the identifier of the Wallet User does match the user bound to the key or key pair* • *the Wallet User is active.* • *the integrity of all operation related assets was verified.* • *the Wallet Provider carrying out operations is authorized and authenticated.*
FDP_ACF.1.3/Operation	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *none.*
FDP_ACF.1.4/Operation	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *any direct key access or key export requests for keys residing in the WSCD shall be denied [assignment: additional rules, based on security attributes, that explicitly deny access of subjects to objects].*

6.4.7 FDP_RIP.2 Full residual information protection

FDP_RIP.2.1	The TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource from all objects.

6.4.8 FDP_SDI.1 Stored data integrity monitoring

FDP_SDI.1.1	The TSF shall monitor user data stored in containers controlled by the TSF **or in its operative environment** for *integrity or authenticity errors* on all objects, based on the following attributes: *the integrity and authenticity check value bound to each stored object.*

6.4.9 FDP_SDC.1 Stored data confidentiality

FDP_SDC.1.1	The TSF shall ensure the confidentiality of the following user data*: session keys and WSCA-stored private keys* while it is stored in the persistent memory (including the external database).

6.5 Class FIA — Identification and Authentication

6.5.1 FIA_AFL.1/Wallet Authentication failure handling

FIA_AFL.1.1/Wallet	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 *the authentication of the wallet user following FIA_UAE_EXT.1*.
FIA_AFL.1.2/Wallet	When the defined number of unsuccessful authentication attempts has been met, the TSF shall *lock the wallet user and notify the wallet user via an out-of-band notification channel*.

Application Note 9

Applicable for verification attempts occur for the same Wallet Instance.

6.5.2 FIA_AFL.1/Provider Authentication failure handling

FIA_AFL.1.1/Provider	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 *the authentication of the wallet provider.*
FIA_AFL.1.2/Provider	When the defined number of unsuccessful authentication attempts has been met, the TSF shall [assignment: list of actions]

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

6.5.3 FIA_AFL.1/Instance Authentication failure handling

FIA_AFL.1.1/Instance	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 *the authentication of the wallet instance.*
FIA_AFL.1.2/Instance	When the defined number of unsuccessful authentication attempts has been met, the TSF shall [assignment: list of actions].

6.5.4 FIA_ATD.1/Wallet User attribute definition

FIA_ATD.1.1/Wallet	The TSF shall maintain the following list of security attributes belonging to individual users: *a) wallet instance and wallet user identifier* *b) the registered Wallet Instance authentication credential, used both for authentication of the communication session and for verification of signatures over User Authentication Evidence* *c) the Authentication Reference Data against which the knowledge- and possession-based factor of the User Authentication Evidence is verified* *d) the binding records associating the wallet user with its wrapped key blobs, its attested key pairs and, where the WIA device-binding key pair is managed by the WSCA/WSCD, that key pair* *e) the per-key authorisation data associated with the wallet user for access to the WSCD-HSM, for the duration of the operation for which it was provided* *f) the authentication failure counter* *g) the wallet user status (active / suspended / deprovisioned)* *h) the integrity check value and version counter bound to the stored attributes of the wallet user* i) *[assignment: additional user security attributes].*

6.5.5 FIA_ATD.1/Provider User attribute definition

FIA_ATD.1.1/Provider	The TSF shall maintain the following list of security attributes belonging to individual users: a) *the Wallet Provider identifier* b) *the Wallet Provider authentication reference data* c) *[assignment: additional provider security attributes]*

6.5.6 FIA_USB.1/Wallet User-subject binding

FIA_USB.1.1/Wallet	The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: • *the wallet instance and wallet user identifier* • *the wallet instance and wallet user authentication data* • *the wallet user status defined in FIA_ATD.1/Wallet* • *the binding records in FIA_ATD.1.1/Wallet* • *the per-key authorisation data in FIA_ATD.1.1/Wallet* • *[assignment: list of user security attributes]*

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

FIA_USB.1.2/Wallet	The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: • *the wallet instance identifier and the registered Wallet Instance authentication credential shall be associated only after successful identification in accordance with FIA_UID.2 and successful establishment of a trusted channel in accordance with FTP_PRO.2/Wallet* • *the wallet user identifier, the wallet user status, the binding records and the per-key authorisation data shall be associated only after successful verification of authentication evidence in accordance with FIA_UAE_EXT.1 against the Authentication Reference Data registered for that wallet user* • *only the attributes of the wallet user for whom the identified wallet instance is registered shall be associated* • *only the binding records and per-key authorisation data for the keys bound to that wallet user shall be associated* • *no attribute shall be associated where the wallet user status is other than active* • *no attribute shall be associated where an integrity or authenticity error has been detected in accordance with FDP_SDI.1 on the stored attributes concerned* • *[assignment: rules for the initial association of attributes]*
FIA_USB.1.3/Wallet	The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: • *the attributes associated under FIA_USB.1.2/Wallet* • *shall remain associated only for the operation for which the authentication evidence was verified, and shall be dissociated on completion of that operation or on expiry of the validity window of that evidence, whichever occurs first;* • *no associated attribute shall be modified at the request of the subject;* • *the authentication failure counter may change as a result of FIA_AFL.1;* • *a change of the wallet user status to suspended or deprovisioned shall take effect immediately and shall cause dissociation of all attributes associated with subjects acting on behalf of that wallet user;* • *a change to the binding records or to the per-key authorisation data shall take effect on the next association only.* • *[assignment: rules for the changing of attributes].*

6.5.7 FIA_USB.1/Provider User-subject binding

FIA_USB.1.1/Provider	The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: • *the wallet provider identifier* • *the wallet provider authentication data* • *[assignment: list of user security attributes]*
FIA_USB.1.2/Provider	The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: • *attributes shall be associated only after successful identification in accordance with FIA_UID.2 and successful establishment of a trusted channel in accordance with FTP_PRO.2/Provider.* • *only the attributes of the Wallet Provider identified by the presented credential shall be associated;* • *only the role and the management functions for which that Wallet Provider is authorised shall be associated.*
FIA_USB.1.3/Provider	The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: • *no associated attribute shall change for the duration of the association;* • *revocation or modification of the Wallet Provider authentication reference data shall cause dissociation of all attributes associated with subjects acting on its behalf;* • *a change to the role or to the permitted management functions shall take effect on the next association only*

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

6.5.8 FIA_UAU.2 User authentication before any action

FIA_UAU.2.1	The TSF shall require each user to be successfully authenticated before allowing any other TSF mediated actions on behalf of that user.

6.5.9 FIA_UAU.6 Re-authenticating

FIA_UAU.6.1	The TSF shall re-authenticate the user under the conditions *before each signing request and for each credential management operation*.

6.5.10 FIA_UID.2 User identification before any action

FIA_UID.2.1	The TSF shall require each user to be successfully identified before allowing any TSF-mediated actions on behalf of that user.

6.5.11 FIA_UAE_EXT.1 Transaction-bound authentication evidence

FIA_UAE_EXT.1.1	The TSF shall generate, for each request to perform an operation in [assignment: list of operations requiring authentication evidence], a challenge with a minimum of 128 bits of entropy obtained from a random bit generation function meeting FCS_RBG.1.
FIA_UAE_EXT.1.2	The TSF shall construct, for each challenge, a transaction context comprising the challenge, the identity of the user, the requested operation, and *the wallet user identifier and [assignment: other transaction attributes]* and shall associate that transaction context with the communication session on which the request was received.
FIA_UAE_EXT.1.3	The TSF shall require that authentication evidence cryptographically bound to the transaction context of the requested operation be presented before the TSF performs any operation in *operations defined in FIA_UAE_EXT.1.1*.
FIA_UAE_EXT.1.4	The TSF shall verify presented authentication evidence against the authentication reference data associated with the identified user and shall verify that the evidence demonstrates that each authentication factor in *a) the possession-based factor of the wallet user and* *b) the knowledge-based factor derived from the PIN* was verified.
FIA_UAE_EXT.1.5	The TSF shall verify that presented authentication evidence was received no more than [assignment: validity period] after the corresponding challenge was issued.
FIA_UAE_EXT.1.6	The TSF shall deny the requested operation if any verification required by FIA_UAE_EXT.1.3, FIA_UAE_EXT.1.4 or FIA_UAE_EXT.1.5 fails.

6.5.12 FIA_UAE_EXT.2 Single-use enforcement for authentication evidence

FIA_UAE_EXT.2.1	The TSF shall maintain a nonce registry covering all transaction identifiers issued and all evidence received within the validity window defined in FIA_UAE_EXT.1.5.
FIA_UAE_EXT.2.2	The TSF shall deny the requested operation if the presented authentication evidence refers to a challenge that was not issued by the TSF, that has already been used in successful verification, or whose validity period has expired.

6.6 Class FMT — Security Management

6.6.1 FMT_SMR.1 Security roles

FMT_SMR.1.1	The TSF shall maintain the roles • *Wallet User* • *Wallet Instance* • *Wallet Provider* • [assignment: additional authorized identified roles].

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

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

6.6.2 FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1	The TSF shall be capable of performing the following management functions:
	• *For the Wallet Provider:*
	  ○ *TOE unique self-identification of version and configuration*
	  ○ *TOE maintenance*
	  ○ *Audit data review*
	  ○ *Factory Reset*
	  ○ *Deletion or reset of user authentication data*
	  ○ *[assignment: list of management functions to be provided by the TSF].*
	• *For the Wallet User:*
	  ○ *Modifying user authentication data*
	  ○ *PIN reset using PUK*
	  ○ *Wallet Instance Migration*

6.6.3 FMT_MSA.1/User_Creation Management of security attributes

FMT_MSA.1.1/ User_Creation	The TSF shall enforce the *User Creation SFP* to restrict the ability to modify the security attributes that are *Wallet User related as defined in FIA_USB.1/Wallet* to *the Wallet Provider or Wallet Instance*.

6.6.4 FMT_MSA.3/User_Creation Static attribute initialization

FMT_MSA.3.1/ User_Creation	The TSF shall enforce the *User Creation SFP* to provide restrictive default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/ User_Creation	The TSF shall allow the *Wallet Provider or Wallet Instance* to specify alternative initial values to override the default values when an object or information is created.

6.6.5 FMT_MSA.1/Key_Management Management of security attributes

FMT_MSA.1.1/ Key_Management	The TSF shall enforce the *Key Management SFP* to restrict the ability to modify, delete the security attributes *Wallet User keys related attributes as defined in FIA_USB.1/Wallet* to *the Wallet Instance or Wallet Provider (deletion only).*

6.6.6 FMT_MSA.3/Key_Management Static attribute initialization

FMT_MSA.3.1/ Key_Management	The TSF shall enforce the *Key Management SFP* to provide restrictive default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/ Key_Management	The TSF shall allow the *Wallet Instance* to specify alternative initial values to override the default values when an object or information is created.

6.6.7 FMT_MSA.1/Operation Management of security attributes

FMT_MSA.1.1/ Operation	The TSF shall enforce the *Operation SFP* to restrict the ability to *query* the security attributes *Wallet User keys related attributes as defined in FIA_USB.1/Wallet* to *the Wallet User*.

6.6.8 FMT_MSA.2 Secure security attributes

FMT_MSA.2.1	The TSF shall ensure that only secure values are accepted for *attributes defined in FIA_USB.1/Wallet and FIA_USB.1/Provider*.

6.6.9 FMT_MTD.1/Provider Management of TSF data

FMT_MTD.1.1/Provider	The TSF shall restrict the ability to modify the *Wallet Provider and WSCD authentication data* to *Wallet Provider*.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

6.6.10 FMT_MTD.1/User Management of TSF data

FMT_MTD.1.1/Provider	The TSF shall restrict the ability to modify ~~the~~ *their Wallet User authentication data* to *an authenticated Wallet User. However, delete and clear are available for the Wallet Provider.*

6.6.11 FMT_MTD.1/Instance Management of TSF data

FMT_MTD.1.1/Provider	The TSF shall restrict the ability to modify ~~the~~ *their Wallet Instance authentication data* to *an authenticated Wallet Instance. However, delete and clear are available for the Wallet Provider.*

6.7 Class FPT — Protection of the TSF

6.7.1 FPT_FLS.1 Failure with preservation of secure state

FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: *WSCD-HSM connection failure; credential store integrity check failure; audit log unavailability; [assignment: additional types of failures in the TSF].*

6.7.2 FPT_ITC.1 Inter-TSF confidentiality during transmission

FPT_ITC.1.1	The TSF shall protect all TSF data transmitted from the TSF to ~~another trusted IT product~~ the WSCD-HSM from unauthorized disclosure during transmission.

Application Note 10

This is applicable for channel with the WSCD-HSM.

6.7.3 FPT_ITI.1 Inter-TSF detection of modification

FPT_ITI.1.1	The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and ~~another trusted IT product~~ the WSCD-HSM within the following metric: [assignment: a defined modification metric].
FPT_ITI.1.2	The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and ~~another trusted IT product~~ the WSCD-HSM and perform [assignment: action to be taken] if modifications are detected.

Application Note 11

The TSF shall detect modification of all data transmitted on the on the WSCA-BE to WSCD-HSM channel. All HSM command responses shall be verified for integrity before use.

6.7.4 FPT_RCV.4 Function recovery

FPT_RCV.4.1	The TSF shall ensure that *all credential management and HSM operations [assignment: additional list of functions and failure scenarios]* have the property that the function either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

Application Note 12

The TSF shall ensure that all credential management and HSM operations are transactional: either they complete fully — including HSM confirmation and audit record creation — or they are rolled back to the prior state, with an audit record of the failure.

6.7.5 FPT_TST.1 TSF self-testing

FPT_TST.1.1	The TSF shall run a suite of the following self-tests during initial start-up, periodically during normal operation, at the request of the authorized user to demonstrate the correct operation of *correct operation of cryptographic libraries if applicable; WSCD-HSM connectivity and authentication; credential store integrity; audit log availability; and correct operation of the transaction identifier registry* [assignment: other parts of the TSF]: [assignment: list of self-tests run by the TSF].

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

FPT_TST.1.2	The TSF shall provide authorized users with the capability to verify the integrity of [selection: [assignment: parts of TSF data], TSF data].
FPT_TST.1.3	The TSF shall provide authorized users with the capability to verify the integrity of [selection: [assignment: parts of TSF], TSF].

6.7.6 FPT_INI.1 TSF initalization

FPT_INI.1.1	The TOE shall provide an initialization function which is self-protected for integrity and authenticity.
FPT_INI.1.2	The TOE initialization function shall ensure that certain properties hold on certain elements immediately before establishing the TSF in a secure initial state, as specified in

ID	Properties	Elements
1	integrity	Software binary, configuration
2	[assignment: property, for instance authenticity, integrity, correct version]	[assignment: list of TSF/user firmware, software or data]

FPT_INI.1.3	The TOE initialization function shall detect and respond to errors and failures during initialization such that the TOE: is halted.
FPT_INI.1.4	The TOE initialization function shall only interact with the TSF in *a restrictive way that allows for the verification of the properties and elements stated in FPT_INI.1.2* during initialization.

6.8 Class FTP — Trusted Path/Channel

6.8.1 FTP_ITC.1/WSCD-HSM Inter-TSF trusted channel

FTP_ITC.1.1/WSCA-HSM	The TSF shall provide a communication channel between itself and ~~another trusted IT product~~ the WSCD-HSM that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2/WSCA-HSM	The TSF shall permit [selection: the TSF, another trusted IT product] to initiate communication via the trusted channel.
FTP_ITC.1.3/WSCA-HSM	The TSF shall initiate communication via the trusted channel for [assignment: list of functions for which a trusted channel is required].

Application Note 13

The TSF shall maintain a mutually authenticated, encrypted channel to the WSCD-HSM providing: identification of the HSM via its hardware attestation certificate; protection against replay of HSM commands using per-command sequence numbers or nonces; and detection of any modification or substitution of HSM responses. The TSF shall refuse to use HSM responses that fail integrity verification.

6.8.2 FTP_PRO.1/Wallet Trusted channel protocol

FTP_PRO.1.1/Wallet	The TSF shall implement [assignment: trusted channel protocol] acting as [assignment: defined protocol role(s)] in accordance with: [assignment: list of standards].
FTP_PRO.1.2/Wallet	The TSF shall enforce usage of the trusted channel for *any communication between the TOE and the Wallet Instance* in accordance with: [assignment: list of standards].
FTP_PRO.1.3/Wallet	The TSF shall permit its peer to initiate communication via the trusted channel.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

FTP_PRO.1.4/Wallet	The TSF shall enforce the following rules for the trusted channel: • *The trusted channel shall be mutually authenticated.* • *[assignment: rules governing operation and use of the trusted channel and/or its protocol].*
FTP_PRO.1.5/Wallet	The TSF shall enforce the following static protocol options: [assignment: list of options and references to standards in which each is defined].
FTP_PRO.1.6/Wallet	The TSF shall negotiate one of the following protocol configurations with its peer: [assignment: list of configurations and reference to standards in which each is defined].

6.8.3 FTP_PRO.2/Wallet Trusted channel establishment

FTP_PRO.2.1/Wallet	The TSF shall establish a shared secret with its peer using one of the following mechanisms: [assignment: list of key establishment mechanisms].
FTP_PRO.2.2/Wallet	The TSF shall authenticate its peer, itself to its peer using one of the following mechanisms: [assignment: list of authentication mechanisms] and according to the following rules: [assignment: list of rules for carrying out the authentication].
FTP_PRO.2.3/Wallet	The TSF shall use [assignment: key derivation function] to derive the following cryptographic keys from a shared secret: [assignment: list of cryptographic keys].

6.8.4 FTP_PRO.3/Wallet Trusted channel data protection

FTP_PRO.3.1/Wallet	The TSF shall protect data in transit from unauthorised disclosure using one of the following mechanisms: [assignment: list of encryption mechanisms].
FTP_PRO.3.2/Wallet	The TSF shall protect data in transit from modification, deletion, insertion, replay, [assignment: other] using one of the following mechanisms: [assignment: list of integrity protection mechanisms].

6.8.5 FTP_PRO.1/Provider Trusted channel protocol

FTP_PRO.1.1/ Provider	The TSF shall implement [assignment: trusted channel protocol] acting as [assignment: defined protocol role(s)] in accordance with: [assignment: list of standards].
FTP_PRO.1.2/ Provider	The TSF shall enforce usage of the trusted channel for *any communication between the TOE and the Wallet Provider* in accordance with: [assignment: list of standards].
FTP_PRO.1.3/ Provider	The TSF shall permit its peer to initiate communication via the trusted channel.
FTP_PRO.1.4/ Provider	The TSF shall enforce the following rules for the trusted channel: [assignment: rules governing operation and use of the trusted channel and/or its protocol]. • *The trusted channel shall be mutually authenticated.* • *[assignment: rules governing operation and use of the trusted channel and/or its protocol].*
FTP_PRO.1.5/ Provider	The TSF shall enforce the following static protocol options: [assignment: list of options and references to standards in which each is defined].
FTP_PRO.1.6/ Provider	The TSF shall negotiate one of the following protocol configurations with its peer: [assignment: list of configurations and reference to standards in which each is defined].

6.8.6 FTP_PRO.2/Provider Trusted channel establishment

FTP_PRO.2.1/ Provider	The TSF shall establish a shared secret with its peer using one of the following mechanisms: [assignment: list of key establishment mechanisms].
FTP_PRO.2.2/ Provider	The TSF shall authenticate its peer, itself to its peer using one of the following mechanisms: [assignment: list of authentication mechanisms] and according to the following rules: [assignment: list of rules for carrying out the authentication].
FTP_PRO.2.3/ Provider	The TSF shall use [assignment: key derivation function] to derive the following cryptographic keys from a shared secret: [assignment: list of cryptographic keys].

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

6.8.7 FTP_PRO.3/Provider Trusted channel data protection

FTP_PRO.3.1/ Provider	The TSF shall protect data in transit from unauthorised disclosure using one of the following mechanisms: [assignment: list of encryption mechanisms].
FTP_PRO.3.2/ Provider	The TSF shall protect data in transit from modification, deletion, insertion, replay, [assignment: other] using one of the following mechanisms: [assignment: list of integrity protection mechanisms].

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

7 Security Assurance Requirements

The assurance package is EAL4 augmented with AVA_VAN.5, ALC_DVS.2 and ALC_FLR.3. The required assurance components and their architecture-specific evaluation focus are defined in ***Table*** 8.

Table 8 — ***Security assurance requirements***

Component	Title	Architecture-specific evaluation notes
ADV_ARC.1	Security Architecture Description	Shall document the WSCA-BE's multi-tenant isolation model, the RAC protocol and the HSM client interface.
ADV_FSP.4	Complete Functional Specification	Shall cover the RAC API endpoint, all HSM command types invoked, all credential store operations and all internal inter-module interfaces.
ADV_IMP.1	Implementation Representation	Source code of the WSCA-BE, including the RAC handler, HSM client library, credential store and audit subsystem.
ADV_TDS.3	Basic Modular Design	Module-level design showing: RAC authentication module; FCS_EXT_UAE evidence verification module; access control engine; HSM client module; credential store; audit module; nonce registry.
AGD_OPE.1	Operational User Guidance	Shall include: HSM configuration and partition setup; WSCA-BE deployment guide; privileged access management requirements; incident response procedures.
AGD_PRE.1	Preparative Procedures	Secure installation and configuration procedures, including HSM initialisation, credential store and integrity baseline creation.
ALC_CMC.4	Production Support and CM Automation	Configuration management system with automated acceptance procedures and production support.
ALC_CMS.4	Problem Tracking CM Coverage	CM coverage for all TOE elements including problem reports and security vulnerabilities.
ALC_DEL.1	Delivery Procedures	Shall cover secure deployment to server infrastructure, including container image signing, SBOM and deployment verification.
ALC_DVS.2	Sufficiency of Security Measures	Shall demonstrate that development and build environment controls are sufficient to protect against supply chain attacks, with evidence of build reproducibility and dependency verification.
ALC_FLR.3	Systematic Flaw Remediation	A flaw remediation process including a responsible disclosure process for server-side vulnerabilities.
ALC_LCD.1	Developer Defined Life-Cycle Model	A documented life-cycle model used in development and maintenance of the WSCA-BE.
ALC_TAT.1	Well-Defined Development Tools	Documented development tools, compiler options and build configurations.
ASE_CCL.1	Conformance Claims	ST conformance claim to this PP.
ASE_ECD.1	Extended Components Definition	Full definition of the FCS_EXT_UAE.1 and FCS_EXT_UAE.2 extended component family, including component hierarchy, management and audit requirements.
ASE_INT.1	ST Introduction	Shall clearly describe the backend deployment model and its relations to wallet instance of user's devices.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Component	Title	Architecture-specific evaluation notes
ASE_OBJ.2	Security Objectives	Full objectives with rationale tracing all threats, OSPs and assumptions to objectives.
ASE_REQ.2	Derived Security Requirements	All SFRs with satisfied dependency analysis, including dependencies of the extended FCS_EXT_UAE components.
ASE_SPD.1	Security Problem Definition	Full SPD including all threats in Clause 4.1.
ASE_TSS.1	TOE Summary Specification	TSS shall describe how the security functions operate, covering the RAC protocol flow, UAE verification, HSM client interaction, multi-tenant isolation and the .
ATE_COV.2	Analysis of Coverage	Shall demonstrate test coverage of all RAC endpoints, all FCS_EXT_UAE evidence verification paths (positive and negative), all multi-tenant isolation boundaries and all HSM error handling paths.
ATE_DPT.2	Testing — Subsystem	Testing at subsystem level of: RAC authentication; FCS_EXT_UAE evidence verification; HSM client; transaction identifier registry. Penetration testing of RAC endpoints shall be included.
ATE_FUN.1	Functional Testing	Functional tests for all TSF functions including evidence verification, multi-tenant isolation and HSM command issuance.
ATE_IND.2	Independent Testing — Sample	Evaluator-performed independent tests with emphasis on FCS_EXT_UAE evidence verification and FDP_ACF.1 multi-tenant access control.
AVA_VAN.5	Advanced Methodical Vulnerability Analysis	Shall include: (a) static analysis of WSCA-BE source code; (b) fuzz testing of all RAC API endpoints; (c) protocol analysis of the RAC authentication and evidence verification protocol; (d) analysis of multi-tenant isolation for cross-wallet-unit leakage paths; (e) HSM client interface analysis for key export and command injection vulnerabilities; (f) analysis of server-side side-channel risks in the deployment environment.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

8 Requirement Objective and Dependency Rationale

8.1 Security Objectives Rationale

Table 9 traces each Threat, OSP and Assumption to the security objectives that address it and provides the rationale for the coverage.

Table 9 — ***Objectives traceability***

Threat / OSP / Assumption	Maps to objective(s)	Rationale
T.WSCA_FW_TAMPER	OE.SECURE_UPDATE OE.VULNERABILITY_MANAGEMENT OE.INFRA_SEC OE.TRUSTED_ADMIN OE.HSM_CERT OT.SELF_PROT OT.KEY_PROTECT OT.KEY_ISOLATION	TOE updates are signed and verified before installation, with an SBOM for each deployed version (OE.SECURE_UPDATE). Compromise of the hosting environment is addressed by change management and intrusion detection (OE.INFRA_SEC). The WSCD-HSM is a certified product (OE.HSM_CERT). Software vulnerability: the developer identifies and remediates vulnerabilities (OE.VULNERABILITY_MANAGEMENT), fixes are delivered through the signed update path (OE.SECURE_UPDATE), a modified binary or configuration is detected at start-up (OT.SELF_PROT), and the least-privilege execution model limits what an exploited process can reach (OE.INFRA_SEC). Insider threat: privileged access management for infrastructure administrators with separation of duties (OE.INFRA_SEC), and trusted administrators, with the residual risk of a malicious administrator stated explicitly (OE.TRUSTED_ADMIN). If the TOE is compromised, its own checks cannot be relied on, and protection rests on controls that do not depend on it: no private key material exists in the TOE (OT.KEY_PROTECT), the WSCD-HSM protects keys as assigned keys usable only with their own authorisation data (OE.HSM_CERT), and that authorisation data is held per key (OT.KEY_ISOLATION). A compromised TOE therefore cannot extract keys, and can use a key only while it holds that key's authorisation data.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Threat / OSP / Assumption	Maps to objective(s)	Rationale
T.WSCA_ATTESTATION_TAMPER	OT.KEY_ISOLATION OE.TRUSTED_WALLET_PROVIDER OE.SECURE_UPDATE OE.VULNERABILITY_MANAGEMENT OE.INFRA_SEC OE.TRUSTED_ADMIN OT.SELF_PROT	Where the attacker submits a public key through the TOE's interfaces, the TOE uses the WSCA/WSCD signing key only for public keys of key pairs generated within the WSCD-HSM and bound to the authenticated wallet user (OT.KEY_ISOLATION), and the Wallet Provider verifies the authenticity of attested public keys before issuing a Key Attestation or Wallet Instance Attestation (OE.TRUSTED_WALLET_PROVIDER). Where the attacker has compromised the TOE, the compromise is prevented as for T.WSCA_FW_TAMPER (OE.SECURE_UPDATE, OE.VULNERABILITY_MANAGEMENT, OE.INFRA_SEC, OE.TRUSTED_ADMIN, OT.SELF_PROT). Compromises are detected by intrusion detection in the operational environment (OE.INFRA_SEC).
T.WSCA_DATA_TAMPER	OT.SELF_PROT OT.PROTECTED_STORAGE OT.AUDIT OE.INFRA_SEC OE.ISOLATION OE.SECURE_UPDATE OE.TRUSTED_ADMIN	Modified binaries or configuration are detected against the baseline and the TOE refuses to start, and HSM credentials are never stored in plaintext (OT.SELF_PROT). Stored data integrity is protected (OT.PROTECTED_STORAGE) and audit records stay tamper-evident (OT.AUDIT). Privilege escalation is limited by the least-privilege execution model and privileged access management for infrastructure administrators (OE.INFRA_SEC) and by workload isolation (OE.ISOLATION), Supply chain compromise by signed updates (OE.SECURE_UPDATE), and insider threat by separation of duties (OE.INFRA_SEC) together with trusted administrators (OE.TRUSTED_ADMIN).
T.WSCA_NET_ATTACK	OT.WSCA_SECURE_CHANNEL OT.USER_AUTH OT.KEY_PROTECT OT.PROTECTED_STORAGE	All traffic is routed through a mutually authenticated channel and each transaction identifier is bound to the Wallet Instance credential (OT.WSCA_SECURE_CHANNEL). Validity window and single use defeat replay (OT.USER_AUTH). The channel credentials are protected assets (OT.KEY_PROTECT) and the session keys are protected at rest (OT.PROTECTED_STORAGE).

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Threat / OSP / Assumption	Maps to objective(s)	Rationale
T.REMOTE_AUTH_VUL	OT.USER_AUTH OT.WSCA_SECURE_CHANNEL OE.UAE_and_ARD OE.RESISTANT_ARD	Evidence must be bound to the registered reference data and to the transaction, be fresh, be single use, and resist tampering and PIN brute force (OT.USER_AUTH). Session binding is tied to the authenticated Wallet Instance (OT.WSCA_SECURE_CHANNEL). The Wallet Instance binds both factors correctly (OE.UAE_and_ARD) and the admissible reference data is constrained (OE.RESISTANT_ARD).
T.WALLET_ORACLE	OE.RESISTANT_ARD OE.UAE_and_ARD OE.TRUSTED_WALLET_USER	The reference data cannot be verified without a secret held by the WSCA-BE or WSCD, so candidates cannot be tested locally and every attempt must pass through the TOE. The admissible reference data is constrained (OE.RESISTANT_ARD). The factors stay protected when the exchange is disclosed (OE.UAE_and_ARD) and the user does not reveal the PIN (OE.TRUSTED_WALLET_USER).
T.WSCD_NET_ATTACK	OT.WSCD_SECURE_CHANNEL OT.SELF_PROT OT.PROTECTED_STORAGE	All traffic to the WSCD is routed through a mutually authenticated channel (OT.WSCD_SECURE_CHANNEL). The HSM access credentials are never held in plaintext (OT.SELF_PROT) and are integrity protected wherever stored outside the WSCD (OT.PROTECTED_STORAGE).
T.MULTI_TENANT	OT.MULTI_TENANT OT.KEY_ISOLATION OT.API_CONTROL	Isolation between wallet users is stated directly (OT.MULTI_TENANT). Each key handle must be bound to the authenticated wallet user before invocation (OT.KEY_ISOLATION) and requests are limited to the caller's own resources (OT.API_CONTROL).
T.KEY_EXTRACT	OT.KEY_PROTECT OT.SELF_PROT OE.HSM_CERT	There is no key material in the TOE to extract, only opaque handles (OT.KEY_PROTECT), and the HSM access credentials are never in plaintext (OT.SELF_PROT). The certified WSCD-HSM resists extraction through its client interface (OE.HSM_CERT).
T.DERIVE	OT.KEY_PROTECT OE.HSM_CERT	Only mandated or ACM-approved algorithms, key lengths and parameters may be used, so public keys and wrapped blobs do not permit derivation (OT.KEY_PROTECT). Key generation is performed by a certified WSCD-HSM (OE.HSM_CERT).

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Threat / OSP / Assumption	Maps to objective(s)	Rationale
T.WSCA_USER_DATA_TAMPER	OT.PROTECTED_STORAGE OT.MULTI_TENANT OE.INFRA_SEC	Integrity and authenticity of key blobs, metadata and user bindings are protected wherever stored (OT.PROTECTED_STORAGE). Wallet users cannot alter one another's records (OT.MULTI_TENANT), and the least-privilege execution model limits a compromised process (OE.INFRA_SEC).
T.PROVIDER_IMPERSONATION	OT.PROVIDER_AUTH OT.WALLET_PROVIDER_SECURE_CHANNEL OT.KEY_PROTECT OT.PROTECTED_STORAGE	The Wallet Provider must authenticate before any provider operation (OT.PROVIDER_AUTH) and all provider traffic runs over a mutually authenticated channel (OT.WALLET_PROVIDER_SECURE_CHANNEL). The provider authentication data is a named protected asset (OT.KEY_PROTECT) and is protected against theft and modification wherever stored (OT.PROTECTED_STORAGE).
T.PROVIDER_NET_ATTACK	OT.PROVIDER_AUTH OT.WALLET_PROVIDER_SECURE_CHANNEL	The Wallet Provider must authenticate before any provider operation (OT.PROVIDER_AUTH) and all provider traffic runs over a mutually authenticated channel (OT.WALLET_PROVIDER_SECURE_CHANNEL).
T.SIDE_CHANNEL	OE.PHYSICAL_PROTECTION OE.ISOLATION	The hardware is hardened against remote side-channel and microarchitectural attacks (OE.PHYSICAL_PROTECTION) and the TOE process is isolated from co-tenants (OE.ISOLATION), removing the precondition.
T.AUDIT_TAMPER	OT.AUDIT OT.PROTECTED_STORAGE OE.AUDIT_LOG_SERVER OE.INFRA_SEC	Records are tamper-evident, integrity and authenticity protected, and carry the identifiers needed for correct attribution (OT.AUDIT). Storage integrity (OT.PROTECTED_STORAGE. The environment receives the records and prevents deletion (OE.AUDIT_LOG_SERVER), with privileged access management covering the insider (OE.INFRA_SEC).
OSP.CRYPTO_STANDARDS	OT.KEY_PROTECT OE.HSM_CERT	The TOE shall use only mandated or ACM-approved algorithms, key lengths and parameters (OT.KEY_PROTECT), executed by a certified WSCD-HSM (OE.HSM_CERT).
OSP.TOE_AUTHENTICATION	OT.AUTHENTICATION	The TOE identifies itself, its version and its configuration uniquely, and authenticates itself to the user (OT. AUTHENTICATION).
OSP.SECURE_ERASURE	OT.SECURE_DELETION	TOE-stored assets are deleted securely and a secure factory reset mechanism is provided, covering the erasure events named in the policy (OT.SECURE_DELETION).

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Threat / OSP / Assumption	Maps to objective(s)	Rationale
OSP.VULN_MGMT	OE.VULNERABILITY_MANAGEMENT OE.SECURE_UPDATE	The developer operates the required vulnerability management process (OE.VULNERABILITY_MANAGEMENT), and updates are signed, verified before installation and accompanied by an SBOM (OE.SECURE_UPDATE).
OSP.INFRA_SEC	OE.INFRA_SEC	The environment provides the required physical and logical controls, including privileged access management with separation of duties, intrusion detection and change management (OE.INFRA_SEC).
OSP.WSCD_CERT	OE.HSM_CERT	The WSCD-HSM is certified against EN 419 221-5 or a different applicable Protection Profile providing at least equivalent functionality and assurance. The private keys must be protected at least to the assigned key definition (OE.HSM_CERT).
OSP.KEY_ISOLATION	OT.KEY_ISOLATION OT.KEY_PROTECT OT.MULTI_TENANT	Keys remain in the WSCD-HSM (OT.KEY_PROTECT) and each handle is bound to the authenticated wallet user identity (OT.KEY_ISOLATION), a binding that cannot be circumvented from another wallet user (OT.MULTI_TENANT).
OSP.AUDIT	OT.AUDIT OE.AUDIT_LOG_SERVER	A tamper-evident record is produced per security-relevant event, attributable to wallet user and transaction (OT.AUDIT), and the environment receives it and prevents deletion (OE.AUDIT_LOG_SERVER).
OSP.MIGRATION	OT.AUDIT OT.USER_AUTH	Key migration is sufficiently logged by OT.AUDIT and Wallet Providers need to be authenticated in OT.USER_AUTH.
A.UAE_and_ARD	OE.UAE_and_ARD OE.GENUINE_WALLET_INSTANCE	The Wallet Instance binds both factors, preserves them when disclosed and uses the device's secure keystore and give requirements on authentication (OE.UAE_and_ARD). It is sourced securely, runs only on secure devices and is certified under (EU) 2024/2981 (OE.GENUINE_WALLET_INSTANCE).
A.PHYSICAL_PROTECTION	OE.PHYSICAL_PROTECTION	The environment protects the TOE against physical attack and hardens the underlying hardware against remote side-channel and microarchitectural attacks (OE.PHYSICAL_PROTECTION
A.TRUSTED_ADMIN	OE.TRUSTED_ADMIN	Administrators are trained, trusted and follow guidance, and the residual risk of a malicious administrator is stated explicitly (OE.TRUSTED_ADMIN).
A.TRUSTED_WALLET_USER	OE.TRUSTED_WALLET_USER	The Wallet User does not disclose their authentication factors (OE.TRUSTED_WALLET_USER).

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Threat / OSP / Assumption	Maps to objective(s)	Rationale
A.TRUSTED_WALLET_PROVIDER	OE.TRUSTED_WALLET_PROVIDER OE.RESISTANT_ARD	Provider personnel are trained and trusted and do not disclose the TOE-related authentication data (OE.TRUSTED_WALLET_PROVIDER); the provider also restricts the admissible reference data (OE.RESISTANT_ARD).
A.AUDIT_LOG_SERVER	OE.AUDIT_LOG_SERVER	A log server in the environment receives the records and ensures log data is not deleted (OE.AUDIT_LOG_SERVER).
A.DATABASE	OE.INFRA_SEC OT.PROTECTED_STORAGE	External databases form part of the secure environment (OE.INFRA_SEC), and the TOE independently protects the integrity and authenticity of data it stores there (OT.PROTECTED_STORAGE).
A.GATEWAY	OE.INFRA_SEC	Gateways form part of the secure environment and provide at least denial-of-service protection (OE.INFRA_SEC).

8.2 Security Functional Requirements Rationale

Table 10 traces each the security objectives to the Security Functional Requirements that address it and provides the rationale for the coverage.

Table 10 ***— Security Functional Requirements Rationale***

Objective	Maps to SFR	Rationale
OT.KEY_PROTECT	FDP_ACC.1/Key_Management FDP_ACF.1/Key_Management FDP_ACC.1/Operation FDP_ACF.1/Operation FCS_CKM.1, FCS_CKM.2 FCS_CKM.5, FCS_COP.1 FTP_ITC.1/WSCD-HSM FMT_MSA.1/Operation FMT_MSA.1/Key_Management FMT_MSA.3/Key_Management FMT_MSA.2	Private keys are generated in and used within the WSCD-HSM and are not part of TOE functionality. The WSCD-HSM is reached by the TOE over a trusted channel (FTP_ITC.1/WSCD-HSM). FDP_ACF.1.4/Operation denies any direct key access or key export request for keys residing in the WSCD, so the TOE holds no unencrypted key material. The Key Management SFP restricts who may create or delete keys, and FMT_MSA.1/Key_Management, FMT_MSA.1/Operation and FMT_MSA.3/Key_Management restrict and initialise the associated attributes. The FCS components cover only operations the TOE performs itself, supported by FMT_MSA.2. Operations performed by the WSCD are referred to the WSCD ST. All cryptographic mechanisms follow OSP.CRYPTO_STANDARDS.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Objective	Maps to SFR	Rationale
OT.KEY_ISOLATION	FDP_ACC.1/Operation FDP_ACF.1/Operation FIA_ATD.1/Wallet FIA_USB.1/Wallet FDP_ACC.1/Key_Creation FDP_ACF.1/Key_Creation	FDP_ACF.1.2/Operation permits an operation only where the identifier of the Wallet User matches the identifier bound to the key or key pair. FIA_ATD.1/Wallet maintains the binding records and the per-key authorisation data FIA_USB.1.2/Wallet associates only the attributes of the wallet user for whom the identified wallet instance is registered, so the attributes of another wallet user can never be associated with the subject. FDP_ACF.1.2/Key_Creation permits the WSCA/WSCD signing key to sign only public keys of key pairs generated within the WSCD-HSM and bound to the identified Wallet User.
OT.SECURE_DELETION	FCS_CKM.6 FDP_ACC.1/Key_Management FDP_ACF.1/Key_Management FMT_SMF.1 FDP_RIP.2	FCS_CKM.6.1 requires destruction of all TSF generated and maintained keys and key material on revocation, deprovisioning, factory reset and authorised deletion. The Key Management SFP (FDP_ACC.1/Key_Management and FDP_ACF.1/Key_Management) restricts who may request deletion, FMT_SMF.1 provides factory reset as a management function, and FDP_RIP.2 ensures that residual content is unavailable on deallocation.
OT.USER_AUTH	FIA_UAE_EXT.1 FIA_UAE_EXT.2 FIA_UID.2 FIA_UAU.2 FIA_UAU.6 FIA_ATD.1/Wallet FIA_USB.1/Wallet FIA_AFL.1 FAU_GEN.1	FIA_UAE_EXT.1 requires a challenge, a transaction context bound to the session, verification of the evidence against the reference data, confirmation that both the possession-based and knowledge-based factors were verified, and denial on any failure. FIA_UAE_EXT.2 enforces single use through the nonce registry. FIA_UAU.6 requires re-authentication before each signing request and each credential management operation. FIA_AFL.1 bounds guessing, FIA_ATD.1/Wallet maintains the reference data and failure counter, FIA_USB.1/Wallet binds them only after successful verification, and FAU_GEN.1 records the outcome.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Objective	Maps to SFR	Rationale
OT.PROVIDER_AUTH	FIA_ATD.1/Provider FIA_USB.1/Provider FIA_UID.2, FIA_UAU.2 FMT_MTD.1/Provider FTP_PRO.2/Provider FIA_AFL.1/Provider	FIA_UID.2 and FIA_UAU.2 require every user, to be identified and authenticated before any action. FIA_ATD.1/Provider maintains the provider identifier and authentication reference data, and FIA_USB.1/Provider associates them only after successful identification and channel establishment, and only for the provider identified by the presented credential. FMT_MTD.1/Provider restricts modification of that reference data to the Wallet Provider.
OT.API_CONTROL	FDP_ACC.1/* FDP_ACF.1/* FIA_UID.2 FIA_UAU.2 FIA_UAU.6 FIA_UAE_EXT.1 FIA_UAE_EXT.2 FIA_ATD.1/Wallet FIA_USB.1/Wallet FIA_AFL.1/Wallet FMT_MSA.1/Operation	FIA_UID.2 and FIA_UAU.2 reject any action before identification and authentication, supported by the authentication mechanism of FIA_UAE_EXT and re-authentication FIA_UAU.6. The three access control policies (FDP_ACC.1, FDP_ACF.1) determine which subject may invoke which operation on which object, and FDP_ACF.1.2/Operation restricts operations to the Wallet User's own keys. FMT_MSA.1/Operation restricts querying of key-related attributes (FIA_ATD.1 and FIA_USB.1) to the Wallet User.
OT.WSCA_SECURE_CHANNEL	FTP_PRO.1/Wallet FTP_PRO.2/Wallet FTP_PRO.3/Wallet FIA_UAE_EXT.1 FIA_AFL.1/Instance	FTP_PRO.1.2/Wallet enforces use of the trusted channel for any communication between the TOE and the Wallet Instance. FTP_PRO.2/Wallet establishes a shared secret and authenticates the peer, and FTP_PRO.3/Wallet protects data in transit from disclosure, modification, deletion, insertion and replay. FIA_UAE_EXT.1.2 associates the transaction context with the communication session on which the request was received, so evidence cannot be presented on another session. **Application Note 14:** In case any cryptographic mechanisms are covered by the TOE itself, the related FCS SFRs shall be mapped here by the ST writer.

Objective	Maps to SFR	Rationale
OT.WSCD_SECURE_CHANNEL	FTP_ITC.1/WSCA FPT_ITC.1 FPT_ITI.1	FTP_ITC.1/WSCD-HSM provides a channel that is logically distinct, assures identification of both endpoints and protects channel data from modification and disclosure. FPT_ITC.1 protects transmitted TSF data from disclosure and FPT_ITI.1 provides detection of modification, with HSM responses verified before use. **Application Note 15:** In case any cryptographic mechanisms are covered by the TOE itself, the related FCS SFRs shall be mapped here by the ST writer.
OT.WALLET_PROVIDER_SECURE_CHANNEL	FTP_PRO.1/Provider FTP_PRO.2/Provider FTP_PRO.3/Provider	FTP_PRO.1.2/Provider enforces use of the trusted channel for any communication between the TOE and the Wallet Provider. FTP_PRO.2/Provider establishes the shared secret and authenticates the peer, and FTP_PRO.3/Provider protects data in transit from disclosure and from modification, deletion, insertion and replay. **Application Note 16:** In case any cryptographic mechanisms are covered by the TOE itself, the related FCS SFRs shall be mapped here by the ST writer.
OT.MULTI_TENANT	FDP_ACC.1/* FDP_ACF.1/* FIA_ATD.1/Wallet FIA_USB.1/Wallet FMT_MSA.1/* FMT_MSA.3/*	The access control policies restrict every controlled operation to the authorised subject and FDP_ACF.1.2/Operation requires the Wallet User identifier to match the identifier bound to the key. FIA_USB.1.2/Wallet prevents the attributes of one wallet user being associated with a subject acting for another and prevents association where the status is other than active or where an integrity error has been detected. FMT_MSA.1 and FMT_MSA.3 restrict and initialise the attributes on which those decisions depend.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Objective	Maps to SFR	Rationale
OT.AUDIT	FAU_GEN.1 FAU_GEN.2 FAU_SAR.1 FAU_STG_EXT.1 FAU_STG_EXT.2 FAU_STG_EXT.3 FCO_NRO.2 FCO_NRR.2	FAU_GEN.1 requires a record for each security-relevant event and FAU_GEN.2 attributes it to the user. FAU_STG_EXT.1.6 requires the authenticity and integrity of audit data to be ensured by cryptographic means, whether held on the TOE or on an external entity, which is the tamper-evidence the objective demands; FAU_STG_EXT.1.1 to .1.5 govern transmission and local retention, FAU_STG_EXT.2 counts losses and FAU_STG_EXT.3 warns before capacity is exceeded. FCO_NRO.2 and FCO_NRR.2 add evidence of origin and receipt verifiable by the Wallet Provider, and FAU_SAR.1 gives the Wallet Provider the means to read the records.
OT.SELF_PROT	FPT_INI.1 FPT_TST.1 FPT_FLS.1 FPT_RCV.4 FDP_RIP.2 FDP_SDC.1	FPT_INI.1 provides a self-protected initialisation function that verifies the integrity of the software binary and configuration and halts on failure. FPT_TST.1 runs self-tests at start-up and periodically over the cryptographic libraries, the WSCD-HSM connection, the credential store and the transaction identifier registry. FPT_FLS.1 preserves a secure state on the named failures and FPT_RCV.4 makes credential management and HSM operations transactional. FDP_RIP.2 prevents recovery of residual content. FDP_SDC.1 ensures the confidentiality of the WSCD-HSM access credentials wherever they are stored, and FDP_RIP.2 ensures that the plaintext copy used in process memory to authenticate to the WSCD-HSM is made unavailable when that memory is deallocated.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Objective	Maps to SFR	Rationale
OT.PROTECTED_STORAGE	FDP_SDI.1 FDP_SDC.1 FDP_ACF.1/Operation FDP_RIP.2 FMT_MTD.1/Instance FMT_MTD.1/User	FDP_SDI.1, FDP_SDC.1 monitors user data stored in containers controlled by the TSF or in its operative environment for integrity, authenticity and confidentiality errors, using the check value bound to each stored object, which is what allows the TOE to rely on external storage without relying on the storage itself. FDP_ACF.1.2/Operation requires the integrity of all operation-related assets to be verified before an operation proceeds. The HSM provides the cryptographic operation used to compute and verify the check values and to protect stored data in confidence. FMT_MTD.1/Instance and FMT_MTD.1/User protect user authentication data from unauthorized modification. FDP_RIP.2 mandates the secure removal of deallocated assets, including potential session keys and per-key authentication data.
OT.AUTHENTICATION	FMT_SMF.1 FMT_SMR.1 FTP_PRO.*	Self-identification is provided by FMT_SMF.1, which requires the TSF to be capable of performing unique self-identification of its version and configuration as a management function available to the users (FMT_SMR.1). Self-Authentication is part of the mutually secure channel FTP_PRO.*.

8.3 Security Dependency Rationale

Table 11 traces dependence for each security functional requirements and provide the rationale for the coverage.

Table 11 ***— Security Dependency Rationale***

SFR	Dependency	Rationale
FAU_GEN.1	FPT_STM.1	The timestamp shall be provided by a reliable time server within the secure operational environment.
FAU_GEN.2	FAU_GEN.1 FIA_UID.1	Covered by FAU_GEN.1 and FIA_UID.1.
FAU_SAR.1	FAU_GEN.1	Covered by FAU_GEN.1.
FAU_STG_EXT.1	FAU_GEN.1	Covered by FAU_GEN.1.
FAU_STG_EXT.2	FAU_GEN.1 FAU_STG_EXT.1	Covered by FAU_GEN.1 and FAU_STG_EXT.1.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

SFR	Dependency	Rationale
FAU_STG_EXT.3	FAU_GEN.1 FAU_STG_EXT.1	Covered by FAU_GEN.1 and FAU_STG_EXT.1.
FCO_NRO.2	FIA_UID.1	Covered by FIA_UID.1.
FCO_NRR.2	FIA_UID.1	Covered by FIA_UID.1.
FCS_CKM.1	[FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1] [FCS_RBG.1 or FCS_RNG.1] FCS_CKM.6	To be defined by the ST author if not covered by the HSM.
FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	To be defined by the ST author if not covered by the HSM.
FCS_CKM.5	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.6	To be defined by the ST author if not covered by the HSM.
FCS_CKM.6	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5]	To be defined by the ST author if not covered by the HSM.
FCS_COP.1	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	To be defined by the ST author if not covered by the HSM.
FCS_RBG.1	[FCS_RBG.2 or FSC_RBG.3] FPT_FLS.1 FPT_TST.1	FCS_RBG.2 or FCS_RBG.3 to be defined by the ST author. Covered by FPT_FLS.1 and FPT_TST.1.
FDP_ACC.1/User_Creation	FDP_ACF.1	Covered by FDP_ACF.1/User_Creation.
FDP_ACF.1/User_Creation	FDP_ACC.1 FMT_MSA.3	Covered by FDP_ACC.1/User_Creation and FMT_MSA.3/User_Creation.
FDP_ACC.1/Key_Management	FDP_ACF.1	Covered by FDP_ACF.1/Key_Management.
FDP_ACF.1/Key_Management	FDP_ACC.1 FMT_MSA.3	Covered by FDP_ACC.1/Key_Management and FMT_MSA.3/Key_Management.
FDP_ACC.1/Operation	FDP_ACF.1	Covered by FDP_ACF.1/Operation.
FDP_ACF.1/Operation	FDP_ACC.1 FMT_MSA.3	Covered by FDP_ACC.1/Operation and FMT_MSA.3/Key_Management (also covers the same attributes as used by the operations).
FDP_RIP.2	n/a	n/a
FDP_SDI.1	n/a	n/a

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

SFR	Dependency	Rationale
FDP_SDC.1	n/a	n/a
FIA_AFL.1	FIA_UAU.1	Covered by FIA_UAU.2 (hierarchical to FIA_UAU.1).
FIA_ATD.1/Wallet	n/a	n/a
FIA_ATD.1/Provider	n/a	n/a
FIA_USB.1/Wallet	FIA_ATD.1	FIA_ATD.1/Wallet
FIA_USB.1/Provider	FIA_ATD.1	FIA_ATD.1/Provider
FIA_UAU.2	FIA_UID.1	Covered by FIA_UID.2 (hierarchical to FIA_UID.1).
FIA_UAU.6	n/a	n/a
FIA_UID.2	n/a	n/a
FIA_UAE_EXT.1	FIA_UID.1 FIA_ATD.1 [FCS_RBG.1 or FCS_RNG.1] FCS_COP.1 FPT_STM1	Covered by FIA_UID.2 (hierarchical to FIA_UID.1), FIA_ATD.1, FCS_RBG.1, FCS_COP.1 functionality is covered by the HSM and the timestamp shall be provided by a reliable time server within the secure operational environment.
FIA_UAE_EXT.2	FPT_STM.1 FIA_UAE_EXT.1	Covered by FIA_UAE_EXT.1 and the timestamp shall be provided by a reliable time server within the secure operational environment.
FMT_SMR.1	FIA_UID.1	Covered by FIA_UID.2 (hierarchical to FIA_UID.1).
FMT_SMF.1	n/a	n/a
FMT_MSA.1/User_Creation	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	Covered by FDP_ACC.1/User_Creation, FMT_SMR.1 and FMT_SMF.1
FMT_MSA.3/User_Creation	FMT_MSA.1 FMT_SMR.1	Covered by FMT_MSA.1/User_Creation and FMT_SMR.1
FMT_MSA.1/Key_Management	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	Covered by FDP_ACC.1/Key_Management, FMT_SMR.1 and FMT_SMF.1
FMT_MSA.3/Key_Management	FMT_MSA.1 FMT_SMR.1	Covered by FMT_MSA.1/Key_Management and FMT_SMR.1
FMT_MSA.1/Operation	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	Covered by FDP_ACC.1/Operation, FMT_SMR.1 and FMT_SMF.1

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

SFR	Dependency	Rationale
FMT_MSA.2	[FDP_ACC.1 or FDP_IFC.1] FMT_MSA.1 FMT_SMR.1	Covered by all FDP_ACC.1/*, FMT_MSA_1/* and FMT_SMR.1
FMT_MTD.1/Provider	FMT_SMR.1 FMT_SMF.1	Covered by FMT_SMR.1 and FMT_SMF.1
FMT_MTD.1/User	FMT_SMR.1 FMT_SMF.1	Covered by FMT_SMR.1 and FMT_SMF.1
FMT_MTD.1/Instance	FMT_SMR.1 FMT_SMF.1	Covered by FMT_SMR.1 and FMT_SMF.1
FPT_FLS.1	n/a	n/a
FPT_ITC.1	n/a	n/a
FPT_ITI.1	n/a	n/a
FPT_RCV.4	n/a	n/a
FPT_TST.1	n/a	n/a
FPT_INI.1	n/a	n/a
FTP_ITC.1/WSCA	n/a	n/a
FTP_PRO.1/Wallet	FTP_PRO.2 FTP_PRO.3	Covered by FTP_PRO.2/Wallet and FTP_PRO.3/Wallet.
FTP_PRO.2/Wallet	FTP_PRO.1 [FCS_CKM.1 or FCS_CKM.2] FCS_CKM.5 FCS_COP.1	Covered by FTP_PRO.1/Wallet, FCS* to be sourced from the HSM or to be covered by the ST writer as required.
FTP_PRO.3/Wallet	FTP_PRO.1 FTP_PRO.2 FCS_COP.1	Covered by FTP_PRO.1/Wallet, FTP_PRO.2/Wallet, FCS* to be sourced from the HSM or to be covered by the ST writer as required.
FTP_PRO.1/Provider	FTP_PRO.2 FTP_PRO.3	Covered by FTP_PRO.2/Provider and FTP_PRO.3/Provider.
FTP_PRO.2/Provider	FTP_PRO.1 [FCS_CKM.1 or FCS_CKM.2] FCS_CKM.5 FCS_COP.1	Covered by FTP_PRO.1/Provider, FCS* to be sourced from the HSM or to be covered by the ST writer as required.
FTP_PRO.3/Provider	FTP_PRO.1 FTP_PRO.2 FCS_COP.1	Covered by FTP_PRO.1/Provider, FTP_PRO.2/Provider, FCS* to be sourced from the HSM or to be covered by the ST writer as required.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

8.4 Security Assurance Requirements Rationale

EAL4 augmented with ALC_DVS.2, ALC_FLR.3 and AVA_VAN.5 is required for this type of product since it is intended to withstand sophisticated attacks.

ALC_DVS.2 covers physical, procedural and personal procedures in place in the development environment. This prevents leakage of design and implementation details that would make attacks easier. ALC_DVS.2 replaces ALC_DVS.1 which is part of the EAL4 package.

ALC_FLR.3 covers flaw remediation handling and does not have any dependencies by itself and can therefore be easily augmented to EAL4.

AVA_VAN.5 is augmented as the TOE shall be highly resistant to any kind of penetration attacks; its dependencies are already fulfilled by EAL4.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

A (informative) Mapping to (EU) 2024/2981 Risk Register

Table 12 maps high-level risks from the risk register mandated by Annex I of Commission Implementing Regulation (EU) 2024/2981 to the Security Problem Definition as defined in chapter 3.

Table 12 ***— Mapping of (EU) 2024/2981 Annex I risk register to Security Problem Definition***

Risk register reference — Annex I	WSCA SPD references	Coverage rationale
R1 — Creation or use of an existing electronic identity	T.WSCA_FW_TAMPER; T.REMOTE_AUTH_VUL; T.WSCA_ATTESTATION_TAMPER; T.MULTI_TENANT. Assets: AST.Attested_Key_Pairs, AST.WIA_Device_Binding_Key_Pair, AST.User_Data.	Addresses misuse of an existing user's keys and identity-related cryptographic operations. Coverage is limited to the WSCA backend; the complete wallet identity creation and onboarding process is outside the defined TOE scope.
R2 — Creation or use of a fake electronic identity	T.WSCA_ATTESTATION_TAMPER; T.WSCA_FW_TAMPER. Assets: AST.Attested_Key_Pairs, AST.WSCA_WSCD_Signing_Key_Pair.	Addresses signing of unauthorised public keys and misuse of attestation-related signing operations. Validation of the complete identity issuance process requires additional wallet-system controls.
R3 — Creation or use of fake attributes	T.WSCA_ATTESTATION_TAMPER; T.WSCA_FW_TAMPER. Assets: AST.Attested_Key_Pairs, AST.WSCA_WSCD_Signing_Key_Pair.	Provides partial coverage where forged or attacker-controlled keys could result in misleading key attestations. Attribute authenticity and issuer validation are **not** explicitly covered by the current SPD and requires additional wallet-system controls.
R4 — Identity theft	T.WSCA_FW_TAMPER; T.REMOTE_AUTH_VUL; T.WALLET_ORACLE; T.MULTI_TENANT; T.WSCA_USER_DATA_TAMPER; T.PROVIDER_IMPERSONATION; T.PROVIDER_NET_ATTACK.	Addresses unauthorised use of user keys, authentication bypass, PIN brute-force attacks, cross-user access and Wallet Provider impersonation.
R5 — Data theft	T.KEY_EXTRACT; T.DERIVE; T.WSCA_NET_ATTACK; T.WSCD_NET_ATTACK; T.SIDE_CHANNEL; T.MULTI_TENANT; T.WSCA_USER_DATA_TAMPER. Assets: AST.Attested_Key_Pairs, AST.WIA_Device_Binding_Key_Pair, AST.User_Data, AST.Wrapped_Key_Blobs.	Addresses extraction, derivation or unauthorised access to sensitive data and cryptographic keys, including through network attacks, side channels and compromised user isolation.
R6 — Data disclosure	T.WSCA_NET_ATTACK; T.WSCD_NET_ATTACK; T.KEY_EXTRACT; T.DERIVE; T.SIDE_CHANNEL; T.WALLET_ORACLE. Assets: AST.Authentication_Reference_Data, AST.WSCD_HSM_Access_Credentials, AST.User_Data, AST.Wrapped_Key_Blobs.	Addresses disclosure through insecure channels, credential exposure, side-channel analysis and attacks against authentication data.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

Risk register reference — Annex I	WSCA SPD references	Coverage rationale
R7 — Data manipulation	T.WSCA_DATA_TAMPER; T.WSCA_ATTESTATION_TAMPER; T.WSCA_USER_DATA_TAMPER; T.WSCA_NET_ATTACK; T.WSCD_NET_ATTACK.	Directly addresses tampering with TOE software, configuration, user data, attestation inputs and communications.
R8 — Data loss	T.WSCA_DATA_TAMPER; T.WSCA_USER_DATA_TAMPER; T.AUDIT_TAMPER. Assets: AST.User_Data, AST.Audit_Records, AST.Wrapped_Key_Blobs.	Addresses unauthorised modification or deletion of protected data and audit records. Availability, backup and recovery requirements are **not** explicitly defined in the current SPD and requires additional wallet-system controls.
R9 — Unauthorised transaction	T.WSCA_FW_TAMPER; T.REMOTE_AUTH_VUL; T.WALLET_ORACLE; T.WSCA_ATTESTATION_TAMPER; T.MULTI_TENANT; T.PROVIDER_IMPERSONATION; T.PROVIDER_NET_ATTACK.	Addresses unauthorised cryptographic operations resulting from authentication bypass, compromised credentials, forged attestation inputs or cross-user access.
R10 — Transaction manipulation	T.WSCA_NET_ATTACK; T.WSCD_NET_ATTACK; T.WSCA_ATTESTATION_TAMPER; T.WSCA_DATA_TAMPER. Assets: AST.Channel_Authentication_Session_Key, AST.Attested_Key_Pairs, AST.User_Data.	Addresses manipulation of communications, attestation inputs and TOE data. Binding cryptographic operations to the complete transaction context is **not** explicitly established by the current SPD and requires additional wallet-system controls.
R11 — Repudiation	T.AUDIT_TAMPER. Asset: AST.Audit_Records. OSP.AUDIT.	Directly addresses suppression, modification, deletion and incorrect attribution of security-relevant audit records. OSP.AUDIT requires tamper-evident records and evidence of operation origin and receipt.
R12 — Transaction data disclosure	T.WSCA_NET_ATTACK; T.WSCD_NET_ATTACK; T.SIDE_CHANNEL; T.WALLET_ORACLE. Assets: AST.Channel_Authentication_Session_Key, AST.Authentication_Reference_Data, AST.User_Data.	Addresses disclosure of sensitive information through communications, side channels and authentication-related attacks. Explicit classification and minimisation of transaction data would strengthen coverage.
R13 — Service disruption	T.WSCA_DATA_TAMPER; T.WSCA_NET_ATTACK; T.WSCD_NET_ATTACK.	Addresses attacks that could compromise service operation or availability. The current threats do **not** explicitly cover denial of service, resource exhaustion, component failure or recovery and requires additional wallet-system controls.
R14 — Surveillance	T.SIDE_CHANNEL; T.WSCA_NET_ATTACK. Assets: AST.User_Data, AST.Audit_Records.	Addresses potential leakage of sensitive information through side channels and network observation. Protection against systematic correlation or tracking of users and transactions is **not** explicitly covered and requires additional wallet-system controls.

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

B (informative) Completeness Analysis of the Security Problem Definition

This appendix provides a completeness analysis of the Security Problem Definition (SPD) of the WSCA-BE Protection Profile in Chapter 3.

The purpose of the analysis is to demonstrate that the identified **assets, subjects, threats, Organisational Security Policies (OSPs), and assumptions** provide sufficient coverage of the security concerns arising from the defined TOE scope, its security functionality, and its operational environment.

The analysis is based on the TOE definition and security functionality described in this Protection Profile. In particular, the analysis considers the WSCA-BE security functions, the assets managed or protected by the TOE, the interactions with the Wallet Instance, Wallet Provider and WSCD-HSM, and the environmental dependencies defined for the TOE.

A security concern is considered covered when it is represented by at least one of the following:

- a **Threat**, where an adverse action by a threat agent must be prevented or detected;
- an **Organisational Security Policy**, where a security requirement is imposed on the TOE or its operation by an applicable policy or regulation; or
- an **Assumption**, where the security property is provided by the operational environment or by another component outside the TOE boundary.

B.1 Coverage of the TOE Security Functionality

The TOE overview, in section 1.2, identifies the WSCA-BE as a security-critical backend component responsible for, among other functions, authentication and authorisation, key lifecycle management, protected cryptographic operations, attestation support, multi-tenant isolation, security audit and self-protection. These functions form the basis for the completeness analysis.

Table 13 maps the principal security functions of the TOE to the corresponding elements of the SPD.

Table 13 **— *TOE security concern / function coverage***

TOE security concern / function	Threat(s)	OSP(s)	Assumption(s)
Protection of WSCA software and configuration	T.WSCA_FW_TAMPER, T.WSCA_DATA_TAMPER	OSP.VULN_MGMT, OSP.TOE_AUTHENTICATION	A.PHYSICAL_PROTECTION, A.TRUSTED_ADMIN
Protection of the WSCA–Wallet Instance channel	T.WSCA_NET_ATTACK	—	A.UAE_and_ARD
Remote User Authentication	T.REMOTE_AUTH_VUL, T.WALLET_ORACLE	—	A.UAE_and_ARD, A.TRUSTED_WALLET_USER
Protection of WSCA–WSCD communication	T.WSCD_NET_ATTACK, T.KEY_EXTRACT, T.WSCA_ATTESTATION_TAMPER	OSP.WSCD_CERT, OSP.KEY_ISOLATION	A.PHYSICAL_PROTECTION

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

TOE security concern / function	Threat(s)	OSP(s)	Assumption(s)
Protection and isolation of Wallet Unit keys	T.MULTI_TENANT, T.KEY_EXTRACT, T.DERIVE, T.WSCA_USER_DATA_TAMPER	OSP.KEY_ISOLATION, OSP.WSCD_CERT	A.DATABASE, A.PHYSICAL_PROTECTION
Wallet Provider authentication	T.PROVIDER_IMPERSONATION	OSP.TOE_AUTHENTICATION	A.TRUSTED_WALLET_PROVIDER
Protection of Wallet User data	T.WSCA_USER_DATA_TAMPER, T.SIDE_CHANNEL	OSP.KEY_ISOLATION	A.DATABASE, A.PHYSICAL_PROTECTION
Audit and accountability	T.AUDIT_TAMPER	OSP.AUDIT	A.AUDIT_LOG_SERVER
Secure erasure and lifecycle termination	—	OSP.SECURE_ERASURE	A.DATABASE, A.PHYSICAL_PROTECTION
Cryptographic algorithms and mechanisms	T.DERIVE, T.KEY_EXTRACT	OSP.CRYPTO_STANDARDS	A.PHYSICAL_PROTECTION
Secure infrastructure operation	T.WSCA_DATA_TAMPER, T.WSCA_FW_TAMPER, T.SIDE_CHANNEL	OSP.INFRA_SEC	A.PHYSICAL_PROTECTION, A.TRUSTED_ADMIN
Protection of the Internet-facing interface	T.WSCA_NET_ATTACK	OSP.INFRA_SEC	A.GATEWAY
Protection of external databases	T.WSCA_USER_DATA_TAMPER, T.WALLET_ORACLE	—	A.DATABASE
Protection of audit infrastructure	T.AUDIT_TAMPER	OSP.AUDIT	A.AUDIT_LOG_SERVER
Identification and authentication of the TOE	T.PROVIDER_IMPERSONATION, T.WSCA_NET_ATTACK	OSP.TOE_AUTHENTICATION	—
WSCD certification and trust	T.KEY_EXTRACT, T.DERIVE, T.WSCD_NET_ATTACK	OSP.WSCD_CERT, OSP.KEY_ISOLATION	A.PHYSICAL_PROTECTION

The analysis shows that each principal security function identified in the TOE description is represented in the SPD by one or more threats, OSPs or environmental assumptions.

B.2 Coverage of Critical Assets

The critical assets identified in section 3.1 include authentication keys and data, ARD, attested key pairs, WIA device-binding keys, WSCA/WSCD signing keys, WSCA software and configuration data, WSCD access credentials, wrapped key blobs, User Data and audit records.

Table 14 verifies that each critical asset has a corresponding security concern in the SPD.

Table 14 ***— Coverage of Critical Assets***

Critical asset	Relevant security concern	SPD coverage
AST.Channel_Authentication_Key_Pair	Unauthorized disclosure, modification or impersonation of the secure channel	T.WSCA_NET_ATTACK

Critical asset	Relevant security concern	SPD coverage
AST.Wallet_Instance_Authentication_Data	Compromise of Wallet Instance authentication	T.WSCA_NET_ATTACK
AST.Channel_Authentication_Session_Key	Disclosure or manipulation of protected communication	T.WSCA_NET_ATTACK
AST.Authentication_Reference_Data	Manipulation, disclosure or abuse during authentication	T.REMOTE_AUTH_VUL, T.WALLET_ORACLE
AST.Wallet_Provider_Authentication_Data	Provider impersonation	T.PROVIDER_IMPERSONATION
AST.Attested_Key_Pairs	Unauthorized use, extraction, derivation or cross-user access	T.WSCA_FW_TAMPER, T.MULTI_TENANT, T.KEY_EXTRACT, T.DERIVE, T.WSCA_USER_DATA_TAMPER
AST.WIA_Device_Binding_Key_Pair	Unauthorized use, extraction or derivation	T.WSCA_FW_TAMPER, T.KEY_EXTRACT, T.DERIVE, T.MULTI_TENANT
AST.WSCA_WSCD_Signing_Key_Pair	Unauthorized signing or key compromise	T.WSCA_FW_TAMPER, T.KEY_EXTRACT, T.DERIVE, T.WSCA_ATTESTATION_TAMPER
AST.WSCA_BE_Software_And_Configuration_Data	Modification or subversion of TOE security functions	T.WSCA_DATA_TAMPER, T.WSCA_FW_TAMPER
AST.WSCD_HSM_Access_Credentials	Unauthorized invocation of WSCD functions	T.WSCA_DATA_TAMPER, T.WSCD_NET_ATTACK, T.KEY_EXTRACT
AST.Wrapped_Key_Blobs	Unauthorized access or manipulation of protected key material	T.DERIVE, T.WSCA_USER_DATA_TAMPER
AST.User_Data	Unauthorized disclosure or modification	T.WSCA_USER_DATA_TAMPER, T.SIDE_CHANNEL
AST.Audit_Records	Suppression, modification, deletion or incorrect attribution	T.AUDIT_TAMPER

Therefore, all assets identified in the SPD have at least one corresponding threat or policy establishing their required protection.

B.3 Coverage of the Operational Environment

The WSCA-BE (TOE) operates as part of the Wallet ecosystem and relies on a number of external entities and supporting components that are outside the TOE boundary. The TOE introduction identifies the **Wallet Instance, User Device, Wallet Provider Backend, Wallet User, Wallet Provider, WSCD-HSM, PID/Attestation Providers, Relying Parties, Access Certificate Infrastructure, gateway servers, external database and external log server** as components or entities interacting with or supporting the TOE.

Table 15 verifies that the security dependencies on these external entities are explicitly represented in the Security Problem Definition. For each external entity or component, the analysis identifies the corresponding **Assumption and/or Organisational Security Policy** that establishes the security conditions on which the TOE relies.

Table 15 — *Coverage of the Operational Environment*

External entity / component	Security dependency of the TOE	Assumption / OSP
Wallet Instance (WI)	The Wallet Instance must securely generate and protect authentication evidence, establish the required secure communication, use secure key storage, and correctly mediate User interaction with the TOE.	A.UAE_and_ARD
Wallet User	The User must protect the knowledge-based authentication factor and must not disclose authentication secrets.	A.TRUSTED_WALLET_USER
Wallet Provider Backend (WPB)	The Wallet Provider must use authenticated communication, protect its authentication data, correctly manage Wallet Units and provide valid authorization requests to the TOE.	A.TRUSTED_WALLET_PROVIDER
Wallet Provider	The Wallet Provider is trusted to protect its authentication data and to operate the Wallet Provider functionality according to the required security procedures.	A.TRUSTED_WALLET_PROVIDER
WSCD-HSM	The WSCD-HSM must protect private keys, prevent unauthorized export or use of protected keys, and provide the required secure cryptographic functionality.	OSP.WSCD_CERT, OSP.KEY_ISOLATION
WSCD-HSM operational environment	The physical and operational environment of the WSCD-HSM must prevent unauthorized physical compromise.	A.PHYSICAL_PROTECTION
PID/Attestation Providers	Providers must correctly authenticate the Wallet Unit, validate WIA/KA information and proof-of-possession, and use the applicable trust infrastructure when issuing credentials.	PID/Attestation Providers are only indirectly connected to the TOE via the Wallet Instance. No assumptions on PID/Attestation Providers are expected in this protection profile for the WSCA-BE.
Relying Parties (RP)	Relying Parties must authenticate using the applicable certificates, correctly verify presentations and use the disclosed information according to the applicable requirements.	Relying parties are only indirectly connected to the TOE via the Wallet Instance. No assumptions on relying parties are expected in this protection profile for the WSCA-BE.
Access Certificate Infrastructure / Trust Infrastructure	The trust infrastructure must provide trustworthy trust anchors and certificate information used by ecosystem entities to authenticate one another.	Keys in the WSCA/WSCD are not part of the Access Certificate Infrastructure / Trust Infrastructure. No assumptions on this infrastruture are expected in this protection profile for the WSCA-BE.
Gateway / Internet-facing gateway	The gateway must protect the TOE from external network attacks and provide the specified network security functions.	A.GATEWAY
External Database	The database storing TOE assets must be located in the secure environment and must provide the required protection for TOE-managed data.	A.DATABASE
External Audit Log Server	The audit server must securely receive audit records and prevent their unauthorized deletion.	A.AUDIT_LOG_SERVER

EUDIW CERTIFICATION SYSTEM MANAGEMENT BOARD NL
Ensuring High Assurance of EUDI Wallets & the NL National eID Scheme

External entity / component	Security dependency of the TOE	Assumption / OSP
Server / hosting infrastructure	The infrastructure hosting the TOE must provide physical protection, controlled privileged access and secure administration.	A.PHYSICAL_PROTECTION, A.TRUSTED_ADMIN, OSP.INFRA_SEC
TOE administrators	Administrators must be trusted, appropriately trained and follow the defined secure administration procedures.	A.TRUSTED_ADMIN

Therefore, operational environment is considered complete with respect to the external entities and components identified in the TOE architecture.