

eSA Security Target of Connected eSE 5.3.4 v1.2 Rev 02

D1641801, version 2.3p

Security Target

REVISION HISTORY

2.2p	10/09/2026	Version creation based on internal document version 2.2.
2.3p	01/10/2026	Update tables references numbering.

TABLE OF CONTENTS

1 ST Introduction 8
1.1 ST reference 8
1.2 TOE reference 8
2 TOE Overview 9
2.1 TOE description 9
2.1.1 TOE type and usage 9
2.1.2 TOE life-cycle 10
2.1.3 Non-TOE HW/SW/FW available to the TOE 13
2.2 TOE scope 14
2.2.1 Physical scope 14
2.2.2 Logical scope 14
3 Conformance Claims 16
3.1 Common Criteria version and conformance with CC part 2 and 3 16
3.2 Assurance package 16
3.3 Protection Profile (PP) conformance claim 16
3.4 Conformance claim rationale 16
3.4.1 Conformity of the TOE Type 17
3.4.2 SPD Consistency 17
3.4.2.1 Assets consistency 17
3.4.2.2 Users and Subjects consistency 18
3.4.2.3 Threats consistency 18
3.4.2.4 Organizational Security Policies consistency 19
3.4.2.5 Assumptions consistency 19
3.4.3 Security Objectives Consistency 19
3.4.3.1 Objective for the TOE consistency 20
3.4.3.2 Objective for Environment consistency 21
3.4.4 Conformity of the Requirement (SFR/SAR) 21
3.4.4.1 SFR consistency 21
3.4.4.2 SAR consistency 24
4 Security Problem Definition 25
4.1 Assets 25
4.2 Users and Subjects 26
4.3 Threats 27
4.4 Organizational Security Policies 29
4.5 Assumptions 30
5 Security Objectives 31
5.1 Security Objectives for the TOE 31

5.2 Security Objectives for the Operational Environment 33
5.3 Security Objectives Rationale 34
5.3.1 Threats 34
5.3.1.1 Unauthorized profile and platform management 34
5.3.1.2 Identity Tampering 35
5.3.1.3 eUICC cloning 36
5.3.1.4 LPAd impersonation 36
5.3.1.5 Unauthorized access to the mobile network 36
5.3.1.6 Second Level Threats 37
5.3.2 Organizational Security Policies 38
5.3.3 Assumptions 38
5.3.4 Rationale tables 38
5.3.4.1 Threats Rationale 38
5.3.4.2 Organizational Security Policies Rationale 40
5.3.4.3 Assumptions Rationale 41
6 Extended Components Definition 43
7 Security Requirements 44
7.1 eUICC Security Functional Requirements 44
7.1.1 Identification and authentication 44
7.1.2 Communication 46
7.1.3 Security Domains 50
7.1.4 Platform Services 52
7.1.5 Security management 53
7.1.6 Mobile Network authentication 58
7.2 Runtime Environment Security Requirements 60
7.2.1 CoreLG Security Functional requirements 60
7.2.1.1 Firewall Policy 60
7.2.1.2 Application Programming Interface 65
7.2.1.3 Card Security Management 75
7.2.1.4 AID Management 76
7.2.2 INSTG Security Functional requirements 78
7.2.3 ADELG Security Functional Requirements 78
7.2.4 RMIG Security Functional Requirements 80
7.2.5 ODELG Security Functional Requirements 80
7.2.6 CARG Security Functional Requirements 81
7.2.7 Global Platform Security Functional requirements 82
7.2.8 Underlying platform IC Security Functional Requirements 95
7.3 Security Functional Requirements Rationale 97
7.3.1 SFRs for eUICC rationale 97

7.3.2 SFRs for Runtime Environment rationale 97
7.3.3 SFRs for underlying platform IC rationale 98
7.3.4 SFRs dependency rationale 98
8 TOE Summary Specification 104
8.1 eUICC security functions 104
8.1.1 GSMA.ProfileManagement 104
8.1.2 GSMA.ECASD 104
8.1.3 GSMA.ISDR 104
8.1.4 GSMA.ISDP 104
8.1.5 GSMA.PPR 104
8.2 Runtime Environment security functions 105
8.2.1 GP.CardContentManagement 105
8.2.2 GP.KeyLoading 105
8.2.3 GP.SecurityDomain 105
8.2.4 GP.SecureChannel 105
8.2.5 GP.GPRegistry 106
8.2.6 GP.OS-UPDATE 106
8.2.7 JCS.APDUBuffer 107
8.2.8 JCS.ByteCodeExecution 107
8.2.9 JCS.Firewall 107
8.2.10 JCS.Package 108
8.2.11 JCS.CryptoAPI 108
8.2.12 JCS.KeyManagement 108
8.2.13 JCS.OwnerPIN 109
8.2.14 JCS.EraseResidualData 109
8.2.15 JCS.OutOfLifeDataUndisclosure 109
8.2.16 JCS.RunTimeExecution 109
8.2.17 JCS.Exception 110
8.2.18 OS.Atomicity 110
8.2.19 OS.MemoryManagement 110
8.3 TSS Rationale 111
8.3.1 eUICC SFRs coverage 111
8.3.2 Runtime Environment SFRs coverage 112
9 Composition with IC 118
9.1 Statement of compatibility – Threats part 118
9.2 Statement of compatibility – OSPs part 119
9.3 Statement of compatibility – Assumptions part 119
9.4 Statement of compatibility – Security objectives for the environment part 119
9.5 Statement of compatibility – Security objectives part 120

9.6 Statement of compatibility – SFRs part 121
10 References, Glossary and Abbreviations 122
10.1 External references 122
10.2 Internal references 123
10.3 Glossary 123
10.4 Abbreviations 124

TABLE OF FIGURES

Figure 1 – Product environment 9
Figure 2 – Connected eSE 5.3.4 v1.2 Platform architecture 10
Figure 3 – TOE life-cycle and actors 11
Figure 4 – Product environment – TOE physical boundaries 14
Figure 5 – TOE logical boundaries 15

TABLE OF TABLES

Table 1 – TOE life-cycle (manufacturing flow) 12
Table 2 – TOE life-cycle (OS update flow) 13
Table 3 – TOE components 14
Table 4 - Assets Consistency table 17
Table 5 - User consistency table 18
Table 6 - Subjects Consistency table 18
Table 7 - Threats Consistency table 19
Table 8 - Organizational Security Policies Consistency table 19
Table 9 - Assumptions Consistency table 19
Table 10 - Security objectives for the TOE consistency table 20
Table 11 - Security objectives for the Operational Environment consistency table 21
Table 12 - Security Functional Requirement consistency table 24
Table 15 - Threats and Security Objectives- Coverage 39
Table 14 - Security Objectives and threats 40
Table 17 - Organizational Security Policies and Security Objectives- Coverage 40
Table 16 - Security Objectives and Organizational Security Policies 41
Table 17 - Assumptions and Security Objectives for the Operational Environment- Coverage 41
Table 18 - Assumptions and Security Objectives for the Operational Environment 42
Table 19 - Runtime environment objectives conversion for SFR rationale. 97
Table 20 – SFRs dependency table 102

All the information provided in this document is provided based on our best knowledge and may change over the time to reflect evolution and/or modification of product features and characteristics.

Thales DIS, its affiliate and representatives accept no duty of care nor liability of any kind whatsoever to any third party, and no responsibility for damages, if any, suffered by any third party as a result of decisions made, or not made, or actions taken, or not taken, based on this document.

Product is certified including preparation, user and administration guidance.

Such guidance defines recommendations explaining how to fulfill security objectives for environment as defined in TOE.

Thales DIS highly recommends following such guidance for secure product deployment.

It is up to the risk manager to check or to rely on evidences that guidance are applied by relevant actors.

Thales DIS will not be held responsible for non-implementation of recommendations and associated consequences.

1 ST INTRODUCTION

1.1 ST reference

The ST identification is the following:

Title:	eSA Security Target of Connected eSE 5.3.4 v1.2 Rev 02
Version:	2.3
Author:	Thales
Reference:	D1641801
Publication date:	01/10/2026

The ST Lite identification is the following:

Title:	eSA Security Target of Connected eSE 5.3.4 v1.2 Rev 02
Version:	2.3p
Author:	Thales
Reference:	D1641801
Publication date:	01/10/2026

1.2 TOE reference

Product name:	Connected eSE 5.3.4 v1.2 Rev 02
Developer:	Thales
TOE name:	Connected eSE 5.3.4 v1.2 Rev 02

The following configuration is available for this TOE

Connected eSE 5.3.4 v1.2 Rev 02 with 10 patches (see [IAD])

Identified with:	553430 (EUICCInfo2) and 000A 0000 (Patch rev)
TOE documentation:	Guidance [GUIDES_Rev02]

TOE hardware part: ST54L (see [ST/IC] and [CERT/IC])

2 TOE OVERVIEW

2.1 TOE description

Connected eSE 5.3.4 v1.2 Rev 02 (called **Connected eSE 5.3.4 Rev 1.2** in the rest of document) is combo eSE/eUICC product addressing the consumer electronics mobile market. Both eSE and eUICC features are isolated logically (via framework) and physically (via interface/protocol).

- As embedded Secure Element (eSE), it ensures the data is stored in a safe place and information is given to only authorized applications and people. It is also a multi-applicative security device, intended to host payment, access control, transport and/or loyalty applications.
- As embedded UICC (eUICC), it provides a platform for remote provisioning and subscription management as defined by the GSMA. The eUICC is a tamper-proof chip and embedded in the device. It also ensures the data is stored in a safe place and information is given to only authorize applications and people.

The TOE is the **eUICC** for Consumer Devices.

The TOE is the **eUICC** open platform with multi-application support, such as Java Card, GlobalPlatform, that implements the GSMA Remote SIM Provisioning (RSP) Architecture for Consumer Devices compliant with the GSMA specifications **[SGP.21] [SGP.22] [SGP.23]** and the Trusted Connectivity Alliance eUICC Profile Package implementing **[EUPP].**

The TOE is able to communicate over [ISO7816] (T=0, T=1) contact protocols.

2.1.1 TOE type and usage

The TOE type is an embedded software implementing eUICC features on secure IC.

The eUICC is an UICC embedded in a consumer device. The eUICC is connected to a given mobile network, by the means of its currently enabled MNO Profile.

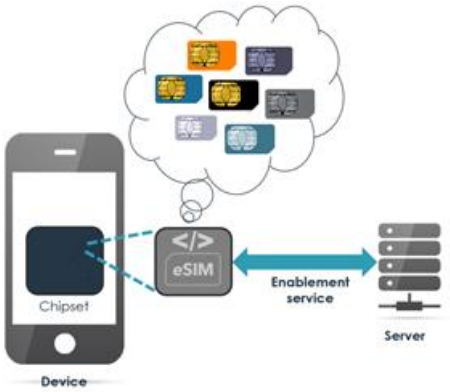


Figure 1 – Product environment

The TOE relies on a Local Profile Assistant (LPA) component. It can either be implemented at the application level as LPAe (the case covered by the LPA PP-Module), or it can be implemented as a non-TOE on-device unit called LPAd. In this product, **the LPAe is not in the TOE**.

The **OS update** capability is available to correct existing features as required by the GSMA specifications.

The **Profiles are not part of the TOE**.

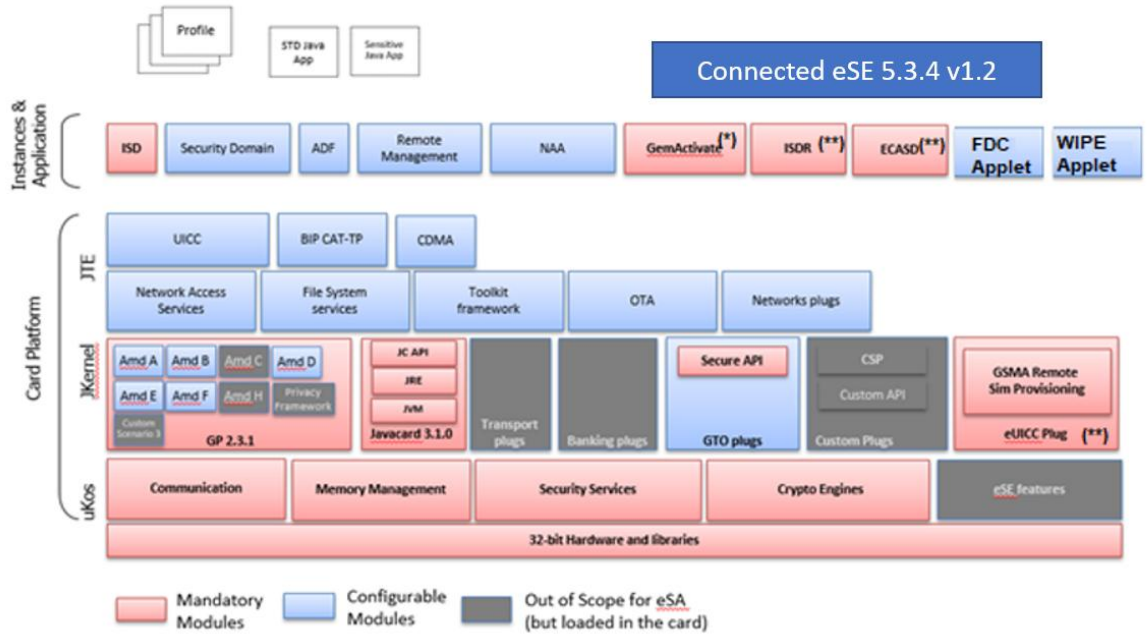


Figure 2 – Connected eSE 5.3.4 v1.2 Platform architecture

Gemactivate (*) is the Advance feature enabler module on the Connected eSE user guide

ISDR (**), ECASD (**) and eUICC plug (**) which are not mandatory on the combo product are mandatory in the scope of this TOE. Configurable modules are optionally present on the product:

- In the "Card Platform", they are activated / deactivated in the embedded software thanks to the GemActivate feature.
- In the "Instances and Applications", they are loaded / instantiated.

The TOE includes 3 layers:

- The hardware layer: ST54L IC providing support to the platform layer
- The platform layer: **Connected eSE 5.3.4 v1.2** OS including the eUICC GSMA Remote Provisioning composed of set of functions providing support to the application layer
- The application layer: composed of ISD-R and ECASD privileged applications providing the remote provisioning and administration functionality, encompassing standard and sensitive applications, as well as the security domains.

2.1.2 TOE life-cycle

The product and TOE lice-cycle is composed of 5 phases (from phase a to e) which are described in Figure 3, in Table 1 and Table 2 with the mention of actors involved in each phase, as well as the associated locations. The TOE delivery is mentioned (dash line in red) before phase d.

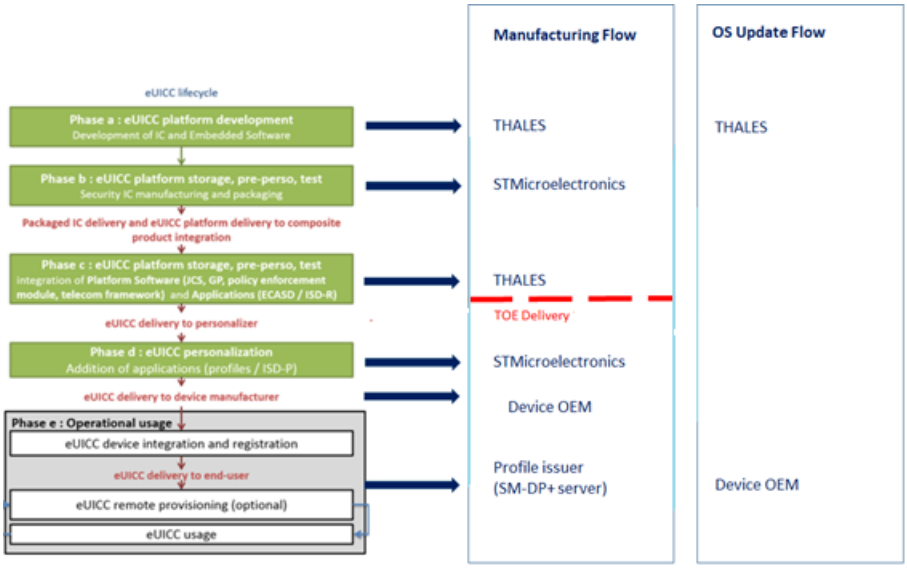


Figure 3 – TOE life-cycle and actors

The actors:

- The **eUICC Manufacturer** (EUM) is the developer of the eUICC secure application (Thales).
- The **IC manufacturer**, STMicroelectronics (STM) is in charge of the **Connected eSE 5.3.4 v1.2** embedded software loading/initialization/pre-personalization and personalization in its own premises and proceeds to the delivery of the product directly to customers.
- The **Device OEM** manufacturer is the Original Equipment Manufacturer
- The **Profile issuer** is MNO that has privilege through its OTA Server to perform Remote Card Content Management (CCM) operations within its own profile (ISD-P). In addition, through its RSP servers, it also can provide Profiles to the end user, but has no privileges to manage profiles remotely without end user consent.
- The **End User** is the user of the device and the eUICC secure application

The manufacturing flow is described in the following table:

Phase	Description	Actor	Location
a	Connected eSE SW (OS and Crypto)	Thales	Thales DIS Singapore site SAS-UP site audit
	ST54L IC development	STMicroelectronics	Development site(s) stated in the ST54L certificate
b	ST54L IC's manufacturing and packaging	STMicroelectronics	Development site(s) stated in the ST54L certificate

c	eUICC OS secure static image build and secure dynamic data generation		
	Product Engineering • Process definition and tools	Thales	THALES DIS Gemenos (France)
	CPC Team • eUICC OS static image preparation	Thales CPC Team	THALES DIS Tczew (Poland)
	Data Generation • Dynamic data generation upon input file reception	Thales PAU Datagen Team	THALES DIS Pont-Audemer (France)
	eUICC OS static image and Dynamic data loading in IC Static and dynamic images are securely delivered to STMicroelectronics with additional transport ciphering thanks to SSH keys used by transfer server (Allynis Connect)	STMicroelectronics	Site(s) stated in the ST54L certificate
*** **TOE delivery** ***			
d	Personalization of data during wafer test flow	STMicroelectronics	Personalisation site(s) stated in the ST54L certificate
Device is delivered to point of sales and is reaching end user			
e	RSP process (Profile loading and activation)	Profile issuer (SM-DP+ server)	In the field Remote access by end user to server associated to MNO / Carrier

Table 1 – TOE life-cycle (manufacturing flow)

The OS update flow is described in the following table (Srikethrough means not applicable in OS Update context):

The conditions to trigger OS update are weakness on eUICC Secure Application (**Connected eSE 5.3.4 v1.2** OS) at security, or functional, or both –OR– deployment of additional feature.

Phase	Description	Actor	Location
a	Connected eSE SW (OS and Crypto)	Thales	Thales DIS Singapore site SAS-UP site audit

***** TOE delivery ***** **Secure delivery of Connected eSE 5.3.4 v1.2 embedded software (patch) to DEVICE EOM**			
e	RSP process (Profile loading and activation)	Profile issuer (SM-DP+ server)	In the field Remote access by end user to the server

Table 2 – TOE life-cycle (OS update flow)

2.1.3 Non-TOE HW/SW/FW available to the TOE

Non-TOE is same than the ones mentioned in the [PP-eUICC] except for IC and RTE.
The TOE does not implement the RMI functions from JCS.

2.2 TOE scope

2.2.1 Physical scope

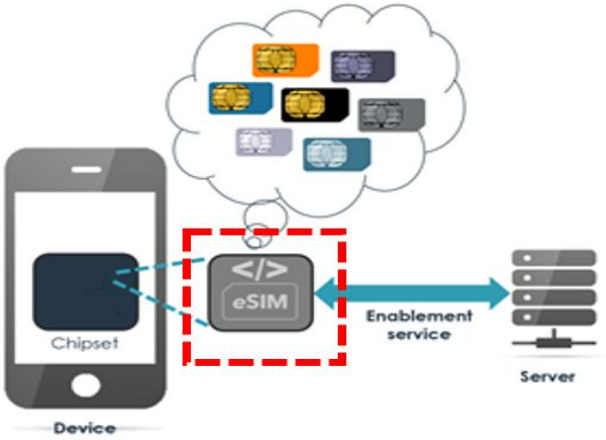


Figure 4 – Product environment – TOE physical boundaries

The physical boundaries encompass the **Connected eSE 5.3.4 v1.2** software executed inside the IC hardware. The other items are outside the scope of the evaluation as illustrated in Figure 4 – Product environment .

The TOE consists of the following components:

TOE component	Developer	Item	Identifier	Form of delivery
IC	STM	ST54L hardware	**ST54L**	Diced wafer (Embedding eUICC OS)
eUICC OS	Thales	Connected eSE 5.3.4 v1.2 Rev 02	553430	Software (Delivered embedded within the IC)
eUICC guidances	Thales	Connected eSE 5.3.4 v1.2 Rev 02	[GUIDES_ Rev02]	Document (Electronic document (PDF) delivered via secure email)

Table 3 – TOE components

2.2.2 Logical scope

The logical boundaries are delimited (dash line in red) in Figure 5.

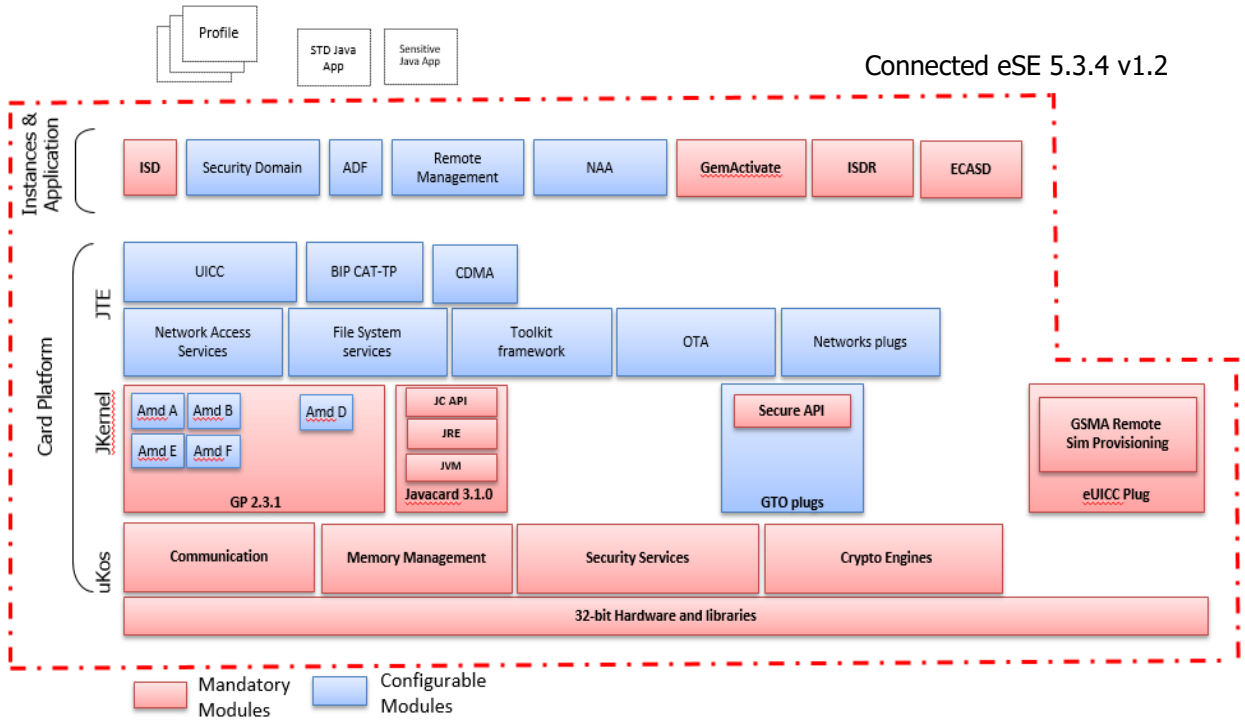


Figure 5 – TOE logical boundaries

The eUICC OS implements the following services:

- Remote Sim Provisioning (RSP) and Local Profile Management (Enable, Disable, Delete MNO Profiles)
- Management and control of the communication between OS and external entities
- OS Security services as:
  - providing secure cryptographic primitives, algorithms and services
  - ensure the security of assets
  - generating random numbers
- Enforcement of the Javacard Runtime and Firewall mechanism
- Standard APIs such as Telecom APIs, JC APIs and GP APIs
- Secure loading of software patches (GemActivate)
  Oracle's Java Card 3.1.0 [JC], which consist of the Java Card 3.1.0 Virtual machine, Java Card Run Time Environment 3.1.0 and the Java Card 3.1.0 Application Programming Interface.
- Global Platform 2.3.1 [11], SE Configuration.
- Secure loading of software patches (GemActivate)
- Support of Cat3 eSIM

3 CONFORMANCE CLAIMS

Evaluation type: This is a composite evaluation, which relies on the ST54L IC certificate and evaluation results.

- Certification done under the NSCIB scheme
- CC certificate: [CERT/IC]
- Security Target [ST/IC] strictly conformance to [PP-84]
- CC version: 3.1, revision 5
- Assurance level: EAL6+ (ASE_TSS.2 augmentation)

The composite evaluation includes the additional composition tasks defined in the [CC-COMP].

3.1 Common Criteria version and conformance with CC part 2 and 3

This Security Target conforms to CC version 3.1 release 5 [CC-1], [CC-2] and [CC-3].

This Security Target is CC Part 2 [CC-2] extended and CC Part 3 [CC-3] conformant of Common Criteria version 3.1, revision 5.

3.2 Assurance package

This Security target conforms to the assurance package EAL4 augmented with ALC_DVS.2 and AVA_VAN.5.

3.3 Protection Profile (PP) conformance claim

This Security Target claims demonstrable conformance to the [PP-eUICC] protection profile.

3.4 Conformance claim rationale

Conformance rationale of the ST against [PP-eUICC] is mapped below. The conformance rationale focuses on assets, threats, OSPs, assumptions, security objectives, and SFRs and the notation used is detailed below:

- Equivalent (E): The element in the ST is the same as in [PP-eUICC].
- Refinement (R): The element in the ST refines the corresponding [PP-eUICC] element. New names are given between brackets and added to the list of elements.
- Addition (A): The element is newly defined in the ST; it is not present in [PP-eUICC] and does not affect it.
- X: The element is present in [PP-eUICC].

3.4.1 Conformity of the TOE Type

The TOE type for this ST is the same as defined in the [PP-eUICC].

The TOE follows the third scenario from the definition in [PP-eUICC], section §1.2.5 when the embedded eUICC is embedded in a certified IC, but the OS and JCS features have not been certified. The ST additionally fulfils the IC objectives and introduces SFRs in order to meet the objectives for the OS and JCS. This is a composite evaluation of the system composed of the eUICC software, JCS and OS on top of a certified IC.

3.4.2 SPD Consistency

3.4.2.1 Assets consistency

All assets defined in [PP-eUICC] are relevant for the TOE of this Security Target. The table below indicates the assets' consistency and the additions from [PP-JCS].

Assets	PP-eUICC	Security Target
D.MNO_KEYS	X	(E)
D.PROFILE_NAA_PARAMS	X	(E)
D.PROFILE_IDENTITY	X	(E)
D.PROFILE_POLICY_RULES	X	(E)
D.PROFILE_USER_CODES	X	(E)
D.PROFILE_CODE	X	(E)
D.TSF_CODE	X	(E)
D.PLATFORM_DATA	X	(E)
D.DEVICE_INFO	X	(E)
D.PLATFORM_RAT	X	(E)
D.SK.EUICC.ECDSA	X	(E)
D.CERT.EUICC.ECDSA	X	(E)
D.PK.CI.ECDSA	X	(E)
D.EID	X	(E)
D.SECRETS	X	(E)
D.CERT.EUM.ECDSA	X	(E)
D.CRLs	X	(E)
D.APP_CODE		(A): Added from [PP-JCS].
D.APP_C_DATA		(A): Added from [PP-JCS].
D.APP_I_DATA		(A): Added from [PP-JCS].
D.APP_KEYs		(A): Added from [PP-JCS].
D.PIN		(A): Added from [PP-JCS].
D.API_DATA		(A): Added from [PP-JCS].
D.CRYPTO		(A): Added from [PP-JCS].
D.JCS_CODE		(A): Added from [PP-JCS].
D.JCS_DATA		(A): Added from [PP-JCS].
D.SEC_DATA		(A): Added from [PP-JCS].
D.OS-UPDATE_SGNVER-KEY		(A): Added from [PP-GP] to cover OS-UPDATE.
D.OS-UPDATE_DEC-KEY		(A): Added from [PP-GP] to cover OS-UPDATE.
D.OS-UPDATE_ADDITIONALCODE		(A): Added from [PP-GP] to cover OS-UPDATE.
D.OS-UPDATE-CODE-ID		(A): Added from [PP-GP] to cover OS-UPDATE.

Table 4 - Assets Consistency table

3.4.2.2 Users and Subjects consistency

All Users defined in [PP-eUICC] are relevant for the TOE of this Security Target. The table below indicates the Users' consistency.

User	PP-eUICC	Security Target
U.SM-DPplus	X	(E)
U.MNO-OTA	X	(E)
U.MNO-SD	X	(E)

Table 5 - User consistency table

All Subjects defined in [PP-eUICC] are relevant for the TOE of this Security Target. The table below indicates the Subjects' consistency and the additions from [PP-JCS].

Subjects	PP-eUICC	Security Target
S.ISD-R	X	(E)
S.ISD-P	X	(E)
S.ECASD	X	(E)
S.PPI	X	(E)
S.PPE	X	(E)
S.TELECOM	X	(E)
S.ADEL		(A): Added from [PP-JCS].
S.APPLET		(A): Added from [PP-JCS].
S.BCV		(A): Added from [PP-JCS].
S.CAD		(A): Added from [PP-JCS].
S.INSTALLER		(A): Added from [PP-JCS].
S.JCRE		(A): Added from [PP-JCS].
S.JCVM		(A): Added from [PP-JCS].
S.LOCAL		(A): Added from [PP-JCS].
S.MEMBER		(A): Added from [PP-JCS].
S.PACKAGE		(A): Added from [PP-JCS].
S.GEMACTIVATE		(A): Added from [PP-GP] to cover OS-UPDATE.

Table 6 - Subjects Consistency table

3.4.2.3 Threats consistency

All Threats defined in [PP-eUICC] are relevant for the TOE of this Security Target. The table below indicates the Threats' consistency.

Threats	PP-eUICC	Security Target
T.UNAUTHORIZED-PROFILE-MNG	X	(R): Assets added from [PP-JCS] are mapped as threatened assets.
T.UNAUTHORIZED-PLATFORM-MNG	X	(R): Assets added from [PP-JCS] are mapped as threatened assets.
T.PROFILE-MNG-INTERCEPTION	X	(R): Assets added from [PP-JCS] are mapped as threatened assets.
T.PROFILE-MNG-ELIGIBILITY	X	(R): Assets added from [PP-JCS] are mapped as threatened assets.
T.UNAUTHORIZED-IDENTITY-MNG	X	(R): Assets added from [PP-JCS] are mapped as threatened assets.

T.IDENTITY-INTERCEPTION	X	(R): Assets added from [PP-JCS] are mapped as threatened assets.
T.UNAUTHORIZED-eUICC	X	(E)
T.LPAd-INTERFACE-EXPLOIT	X	(E)
T.UNAUTHORIZED-MOBILE-ACCESS	X	(E)
T.LOGICAL-ATTACK	X	(R): Assets added from [PP-JCS] are mapped as threatened assets.
T.PHYSICAL-ATTACK	X	(E)
T.UNAUTHORISED-TOE-CODE-UPDATE		(A): Added from [PP-GP] to cover OS-UPDATE.
T.FAKE-SGNVER-KEY		(A): Added from [PP-GP] to cover OS-UPDATE.
T.WRONG-UPDATE-STATE		(A): Added from [PP-GP] to cover OS-UPDATE.
T.INTEG-OS-UPDATE-LOAD		(A): Added from [PP-GP] to cover OS-UPDATE.
T.CONFID-OS-UPDATE-LOAD		(A): Added from [PP-GP] to cover OS-UPDATE.

Table 7 - Threats Consistency table

3.4.2.4 Organizational Security Policies consistency

All Organizational Security Policies defined in [PP-eUICC] are relevant for the TOE of this Security Target. The table below indicates the Organizational Security Policies' consistency.

OSPs	PP-eUICC	Security Target
OSP.LIFE-CYCLE	X	(E)
OSP.ATOMIC_ACTIVATION		(A): Added from [PP-GP] to cover OS-UPDATE.
OSP.TOE_IDENTIFICATION		(A): Added from [PP-GP] to cover OS-UPDATE.
OSP.ADDITIONAL_CODE_SIGNING		(A): Added from [PP-GP] to cover OS-UPDATE.
OSP.ADDITIONAL_CODE_ENCRYPTION		(A): Added from [PP-GP] to cover OS-UPDATE.

Table 8 - Organizational Security Policies Consistency table

3.4.2.5 Assumptions consistency

All Assumptions defined in [PP-eUICC] are relevant for the TOE of this Security Target. The table below indicates the Assumptions consistency.

Assumptions	PP-eUICC	Security Target
A.TRUSTED-PATHS-LPAd	X	(E)
A.ACTORS	X	(E)
A.APPLICATIONS	X	(E)
A.OS-UPDATE-EVIDENCE		(A): Added from [PP-GP] to cover OS-UPDATE.
A.SECURE_ACODE_ MANAGEMENT		(A): Added from [PP-GP] to cover OS-UPDATE.

Table 9 - Assumptions Consistency table

3.4.3 Security Objectives Consistency

3.4.3.1 Objective for the TOE consistency

All Security Objectives defined in [PP-eUICC] are relevant for the TOE of this Security Target. The table below indicates the Security Objectives' consistency.

Note that OE.RE* and OE.IC* from [PP-eUICC] become security objectives from the TOE in the present security target. The [PP-eUICC] already provides the conversion of OE.RE* to objectives from the [PP-JCS] protection profile.

O.TOE	PP-eUICC	Security Target
O.PPE-PPI	X	(E)
O.eUICC-DOMAIN-RIGHTS	X	(E)
O.SECURE-CHANNELS	X	(E)
O.INTERNAL-SECURE-CHANNELS	X	(E)
O.PROOF_OF_IDENTITY	X	(E)
O.OPERATE	X	(E)
O.API	X	(E)
O.DATA-CONFIDENTIALITY	X	(E)
O.DATA-INTEGRITY	X	(E)
O.ALGORITHMS	X	(E)
O.IC.PROOF_OF IDENTITY		(A): Added and replace OE.IC.PROOF_OF IDENTITY from [PP-eUICC].
O.IC.SUPPORT		Added and replace OE.IC.SUPPORT from [PP-eUICC].
O.IC.RECOVERY		(A): Added and replace OE.IC.RECOVERY from [PP-eUICC].
O.RE.PPE-PPI		(A): Added and replace OE.RE.PPE-PPI from [PP-eUICC].
O.RE.SECURE-COMM		Added and replace OE.RE.SECURE-COMM from [PP-eUICC].
O.RE.API		(A): Added and replace OE.RE.API from [PP-eUICC].
O.RE.DATA-CONFIDENTIALITY		(A): Added and replace OE.RE.DATA-CONFIDENTIALITY from [PP-eUICC].
O.RE.DATA-INTEGRITY		(A): Added and replace OE.RE.DATA-INTEGRITY from [PP-eUICC].
O.RE.IDENTITY		(A): Added and replace OE.RE.IDENTITY from [PP-eUICC].
O.RE.CODE-EXE		(A): Added and replace OE.RE.CODE-EXE from [PP-eUICC].
O.SECURE_LOAD_ACODE		(A): Added from [PP-GP] to cover OS-UPDATE.
O.SECURE_AC_ACTIVATION		(A): Added from [PP-GP] to cover OS-UPDATE.
O.TOE_IDENTIFICATION		(A): Added from [PP-GP] to cover OS-UPDATE.
O.CONFID-OS-UPDATE.LOAD		(A): Added from [PP-GP] to cover OS-UPDATE.

Table 10 - Security objectives for the TOE consistency table

3.4.3.2 Objective for Environment consistency

O.ENV	PP-eUICC	Security Target
OE.CI	X	(E)
OE.SM-DPplus	X	(E)
OE.MNO	X	(E)
OE.TRUSTED-PATHS-LPAd	X	(E)
OE.APPLICATIONS	X	(E)
OE.VERIFICATION	X	(A) : added from [PP-JCS]
OE.CODE-EVIDENCE	X	(A):added from [PP-JCS]
OE.MNO-SD	X	(E)
OE.IC.PROOF_OF_IDENTITY	X	Removed and replaced by O.IC.PROOF_OF IDENTITY.
OE.IC.SUPPORT	X	Removed and replaced by O.IC.SUPPORT.
OE.IC.RECOVERY	X	Removed and replaced by O.IC.RECOVERY.
OE.RE.PPE-PPI	X	Removed and replaced by O.RE.PPE-PPI.
OE.RE.SECURE-COMM	X	Removed and replaced by O.RE.SECURE-COMM.
OE.RE.API	X	Removed and replaced by O.RE.API.
OE.RE.DATA-CONFIDENTIALITY	X	Removed and replaced by O.RE.DATA-CONFIDENTIALITY.
OE.RE.DATA-INTEGRITY	X	Removed and replaced by O.RE.DATA-INTEGRITY
OE.RE.IDENTITY	X	Removed and replaced by O.RE.IDENTITY
OE.RE.CODE-EXE	X	Removed and replaced by O.RE.CODE-EXE
OE.OS-UPDATE-EVIDENCE		(A): Added from [PP-GP] to cover OS-UPDATE.
OE.OS-UPDATE-ENCRYPTION		(A): Added from [PP-GP] to cover OS-UPDATE.
OE.SECURE_ACODE_MANAGEMENT		(A): Added from [PP-GP] to cover OS-UPDATE.

Table 11 - Security objectives for the Operational Environment consistency table

3.4.4 Conformity of the Requirement (SFR/SAR)

3.4.4.1 SFR consistency

SFR	PP-eUICC	Security Target
FIA_UID.1/EXT	X	
FIA_UAU.1/EXT	X	
FIA_USB.1/EXT	X	
FIA_UAU.4/EXT	X	
FIA_UID.1/MNO-SD	X	
FIA_USB.1/MNO-SD	X	
FIA_ATD.1	X	

FIA_API.1	X	
FDP_IFC.1/SCP	X	
FDP_IFF.1/SCP	X	
FTP_ITC.1/SCP	X	
FDP_ITC.2/SCP	X	
FPT_TDC.1/SCP	X	
FDP_UCT.1/SCP	X	
FDP_UIT.1/SCP	X	
FCS_CKM.1/SCP-SM	X	
FCS_CKM.2/SCP-MNO	X	
FCS_CKM.4/SCP-SM	X	
FCS_CKM.4/SCP-MNO	X	
FDP_ACC.1/ISDR	X	
FDP_ACF.1/ISDR	X	
FDP_ACC.1/ECASD	X	
FDP_ACF.1/ECASD	X	
FDP_IFC.1/Platform_services	X	
FDP_IFF.1/Platform_services	X	
FPT_FLS.1/Platform_services	X	
FCS_RNG.1	X	
FPT_EMS.1	X	
FDP_SDI.1	X	
FDP_RIP.1	X	
FPT_FLS.1	X	
FMT_MSA.1/PLATFORM_DATA	X	
FMT_MSA.1/PPR	X	
FMT_MSA.1/CERT_KEYS	X	
FMT_SMF.1	X	
FMT_SMR.1	X	
FMT_MSA.1/RAT	X	
FMT_MSA.3	X	
FCS_COP.1/Mobile_network	X	
FCS_CKM.2/Mobile_network	X	
FCS_CKM.4/Mobile_network	X	
FDP_ACC.2/FIREWALL		(A): Added from [PP-JCS].
FDP_ACF.1/FIREWALL		(A): Added from [PP-JCS].
FDP_IFC.1/JCVM		(A): Added from [PP-JCS].
FDP_IFF.1/JCVM		(A): Added from [PP-JCS].
FDP_RIP.1/OBJECTS		(A): Added from [PP-JCS].
FMT_MSA.1/JCRE		(A): Added from [PP-JCS].
FMT_MSA.1/JCVM		(A): Added from [PP-JCS].
FMT_MSA.2/FIREWALL_JCVM		(A): Added from [PP-JCS].
FMT_MSA.3/FIREWALL		(A): Added from [PP-JCS].
FMT_MSA.3/JCVM		(A): Added from [PP-JCS].
FMT_SMF.1/JC		(A): Added from [PP-JCS] and refined
FMT_SMR.1/JC		(A): Added from [PP-JCS] and refined
FCS_CKM.1/ECDSA FCS_CKM.1/GP-SCP		(A): Added from [PP-JCS]. Refined with iteration. (A): Added from [PP-GP].
FCS_CKM.4		(A): Added from [PP-JCS].
FCS_COP.1/TDES_MAC FCS_COP.1/AES_MAC FCS_COP.1/ECDH FCS_COP.1/CRC		(A): Added from [PP-JCS]. Refined with iteration.

FCS_COP.1/ECDSA_SIGN FCS_COP.1/ECKA_EG FCS_COP.1/GP-SCP FCS_COP.1/TDES_CIPHER FCS_COP.1/AES_CIPHER FCS_COP.1/RSA_SIGN FCS_COP.1/RSA_CIPHER FCS_COP.1/Hash FCS_COP.1/HMAC		
FDP_RIP.1/ABORT		(A): Added from [PP-JCS].
FDP_RIP.1/APDU		(A): Added from [PP-JCS].
FDP_RIP.1/bArray		(A): Added from [PP-JCS].
FDP_RIP.1/GlobalArray		(A): Added from [PP-JCS].
FDP_RIP.1/KEYS		(A): Added from [PP-JCS].
FDP_RIP.1/TRANSIENT		(A): Added from [PP-JCS].
FDP_ROL.1/FIREWALL		(A): Added from [PP-JCS].
FAU_ARP.1		(A): Added from [PP-JCS].
FDP_SDI.2/DATA		(A): Added from [PP-JCS].
FPR_UNO.1		(A): Added from [PP-JCS].
FPT_FLS.1/JC		(A): Added from [PP-JCS]. Refined with iteration.
FPT_TDC.1		(A): Added from [PP-JCS].
FIA_ATD.1/AID		(A): Added from [PP-JCS].
FIA_UID.2/AID		(A): Added from [PP-JCS].
FIA_USB.1/AID		(A): Added from [PP-JCS].
FMT_MTD.1/JCRE		(A): Added from [PP-JCS].
FMT_MTD.3/JCRE		(A): Added from [PP-JCS].
FDP_ACC.2/ADEL		(A): Added from [PP-JCS].
FDP_ACF.1/ADEL		(A): Added from [PP-JCS].
FDP_RIP.1/ADEL		(A): Added from [PP-JCS].
FMT_MSA.1/ADEL		(A): Added from [PP-JCS].
FMT_MSA.3/ADEL		(A): Added from [PP-JCS].
FMT_SMF.1/ADEL		(A): Added from [PP-JCS].
FMT_SMR.1/ADEL		(A): Added from [PP-JCS].
FPT_FLS.1/ADEL		(A): Added from [PP-JCS].
FDP_RIP.1/ODEL		(A): Added from [PP-JCS].
FPT_FLS.1/ODEL		(A): Added from [PP-JCS].
FPT_FLS.1/GP		(A): Added from [PP-GP].
FDP_ROL.1/GP		(A): Added from [PP-GP].
FCO_NRO.2/GP		(A): Added from [PP-GP].
FMT_SMR.1/GP		(A): Added from [PP-GP].
FMT_SMF.1/GP		(A): Added from [PP-GP].
FDP_ITC.2/GP-ELF		(A): Added from [PP-GP].
FDP_ITC.2/GP-KL		(A): Added from [PP-GP].
FPT_RCV.3/GP		(A): Added from [PP-GP].
FDP_IFC.2/GP-ELF		(A): Added from [PP-GP].
FDP_IFF.1/GP-ELF		(A): Added from [PP-GP].
FIA_UID.1/GP		(A): Added from [PP-GP].
FIA_AFL.1/GP		(A): Added from [PP-GP].
FIA_UAU.1/GP		(A): Added from [PP-GP].
FIA_UAU.4/GP		(A): Added from [PP-GP].
FDP_UIT.1/GP		(A): Added from [PP-GP].
FDP_UCT.1/GP		(A): Added from [PP-GP].
FTP_ITC.1/GP		(A): Added from [PP-GP].

FPR_UNO.1/GP		(A): Added from [PP-GP].
FPT_TDC.1/GP		(A): Added from [PP-GP].
FDP_IFC.2/GP-KL		(A): Added from [PP-GP].
FDP_IFF.1/GP-KL		(A): Added from [PP-GP].
FMT_MSA.1/GP		(A): Added from [PP-GP].
FMT_MSA.3/GP		(A): Added from [PP-GP].
FDP_ACC.1/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FDP_ACF.1/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FMT_MSA.3/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FMT_SMR.1/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FMT_SMF.1/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FIA_ATD.1/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FTP_TRP.1/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FCS_COP.1/OS-UPDATE-DEC		(A): Added from [PP-GP] to cover OS update.
FCS_COP.1/OS-UPDATE-VER		(A): Added from [PP-GP] to cover OS update.
FPT_FLS.1/OS-UPDATE		(A): Added from [PP-GP] to cover OS update.
FAU_SAS.1		(A): Added to cover O.IC.PROOF_OF_IDENTITY.
FPT_RCV.3/OS		(A): Added to cover O.IC.RECOVERY.
FPT_RCV.4/OS		(A): Added to cover O.IC.SUPPORT.

Table 12 - Security Functional Requirement consistency table

3.4.4.2 SAR consistency

This ST claims the same evaluation assurance level with same rationale as [PP-eUICC], i.e., EAL4 augmented with ALC_DVS.2 and AVA_VAN.5.

4 SECURITY PROBLEM DEFINITION

This chapter introduces the security problem addressed by the TOE and its operational environment. The security problem consists of the threats the TOE may face in the field, the assumptions on its operational environment, and the organizational policies that must be implemented by the TOE or within the operational environment.

4.1 Assets

The definition of the assets from [PP-eUICC] and [PP-JCS] is not repeated here. See section 3.4.2.1 for complete list is assets.

Assets	
D.MNO_KEYS	
D.PROFILE_NAA_PARAMS	
D.PROFILE_IDENTITY	
D.PROFILE_POLICY_RULES	
D.PROFILE_USER_CODES	
D.PROFILE_CODE	
D.TSF_CODE	
D.PLATFORM_DATA	
D.DEVICE_INFO	
D.PLATFORM_RAT	
D.SK.EUICC.ECDSA	
D.CERT.EUICC.ECDSA	
D.PK.CI.ECDSA	
D.EID	
D.SECRETS	
D.CERT.EUM.ECDSA	
D.CRLs	
D.APP_CODE	
D.APP_C_DATA	
D.APP_I_DATA	
D.APP_KEYs	
D.PIN	
D.API_DATA	
D.CRYPTO	
D.JCS_CODE	
D.JCS_DATA	
D.SEC_DATA	
Additional D.Asset to cover OS-UPDATE	
D.OS-UPDATE_SGNVER-KEY	Refinement of D.APP_KEYS. A symmetric cryptographic key, owned by the OS Developer, and used by the TOE to verify the signature of the additional code to be loaded. To be protected from unauthorized disclosure and modification.
D.OS-UPDATE_DEC-KEY	Refinement of D.APP_KEYS. A symmetric cryptographic key, owned by the OS Developer, and used by the TOE to decrypt the additional code to be loaded. To be protected from unauthorized disclosure and modification.

D.OS-UPDATE_ADDITIONALCODE	The code to be added to the OS after TOE issuance. The additional code has to be signed by the OS Developer. After successful verification of the signature by the Initial TOE, the additional code is loaded and installed through an atomic activation (to create an Updated TOE). To be protected from unauthorized disclosure and modification.
D.OS-UPDATE-CODE-ID	The identification data associated with the additional code. It is loaded and/or updated in the same atomic operation as additional code loading. To be protected from unauthorized modification.

4.2 Users and Subjects

The definition of users and subjects from [PP-eUICC] and [PP-JCS] is not repeated here. See section 3.4.2.2 for complete list is users and subjects.

User
U.SM-DPplus
U.MNO-OTA
U.MNO-SD

Subjects	
S.ISD-R	
S.ISD-P	
S.ECASD	
S.PPI	
S.PPE	
S.TELECOM	
S.ADEL	
S.APPLET	
S.BCV	
S.CAD	
S.INSTALLER	
S.JCRE	
S.JCVM	
S.LOCAL	
S.MEMBER	
S.CAP_FILE	
S.PACKAGE	S.PACKAGE and S.CAP_FILE are the same subject
Additional S.Subject to cover OS-UPDATE	
S.GEMACTIVATE	GemActivate Security Domain representing a Thales administrator on the card. This entity can authorize the activation of optional services and the loading of additional code (i.e. patch) post issuance. Note: this subject corresponds to 'S.OS-DEVELOPER' in the PP-Module 'OS Update' of [PP-GP]. S.GEMACTIVATE and S.OS-DEVELOPER are aliases of the same subject.

4.3 Threats

The definition of threats from [PP-eUICC] where no refinements are made is not repeated here. See section 3.4.2.3 for complete list is threats.

The definition of each threat is present in [PP-eUICC]. The mapping against assets has been refined in the column 'refined threats description" where assets in bold come from [PP-JCS] and/or [PP-GP]

Threats	Refined threats description
T.UNAUTHORIZED-PROFILE-MNG	Directly threatens the assets: D.ISDP_KEYS, D.MNO_KEYS, D.TSF_CODE (ISD-P), D.PROFILE_*, **D.APP_C_DATA, D.APP_I_DATA, D.PIN, D.APP_KEYs and D.APP_CODE**.
T.UNAUTHORIZED-PLATFORM-MNG	Directly threatened assets are D.TSF_CODE, D.PLATFORM_DATA, D.PLATFORM_RAT. By altering the behavior of ISD-R or PPE, the attacker indirectly threatens the provisioning status of the eUICC, thus also threatens the same assets as T.UNAUTHORIZED-PROFILE-MNG.
T.PROFILE-MNG-INTERCEPTION	Directly threatens the assets: D.MNO_KEYS, D.TSF_CODE (ISD-P), D.PROFILE_*, **D.APP_C_DATA, D.PIN and D.APP_KEYs**.
T.PROFILE-MNG-ELIGIBILITY	Directly threatens the assets: D.TSF_CODE, D.DEVICE_INFO, D.EID, **D.APP_C_DATA, D.PIN, D.APP_KEYs, D.APP_CODE and D.APP_I_DATA**.
T.UNAUTHORIZED-IDENTITY-MNG	Directly threatens the assets: D.TSF_CODE, D.SK.EUICC.ECDSA, D.SECRETS, D.CERT.EUICC.ECDSA, D.PK.CI.ECDSA, D.EID, D.CERT.EUM.ECDSA, D.CRLs, **D.APP_CODE, D.APP_I_DATA, D.PIN, D.APP_KEYs, D.APP_C_DATA and D.SEC_DATA**.
T.IDENTITY-INTERCEPTION	Directly threatens the assets: D.SECRETS, D.EID, **D.APP_C_DATA, D.PIN and D.APP_KEYs**.
T.UNAUTHORIZED-eUICC	
T.LPAd-INTERFACE-EXPLOIT	
T.UNAUTHORIZED-MOBILE-ACCESS	
T.LOGICAL-ATTACK	Directly threatens the assets: D.TSF_CODE, D.PROFILE_NAA_PARAMS, D.PROFILE_POLICY_RULES, D.PLATFORM_DATA, D.PLATFORM_RAT, **D.JCS_CODE, D.API_DATA, D.SEC_DATA, D.JCS_DATA, D.CRYPTO, D.APP_CODE, D.APP_I_DATA, D.PIN, D.APP_KEYs and D.APP_C_DATA**.
T.PHYSICAL-ATTACK	
Additional T.Threat to cover OS-UPDATE	
T.UNAUTHORISED-TOE-CODE-UPDATE	Threat agent: Attacker Adverse action: An attacker loads malicious additional code in order to compromise the security features of the TOE.

	Directly threatened asset(s): D.OS-UPDATE_ADDITIONALCODE, D.JCS_CODE, D.JCS_DATA.
T.FAKE-SGNVER-KEY	Threat agent: Attacker Adverse action: An attacker modifies the signature verification key used by the TOE to verify the signature of the additional code. Hence, the attacker is able to sign and successfully load malicious additional code inside the TOE. Directly threatened asset(s): D.OS-UPDATE_SGNVER-KEY, D.OS UPDATE_ADDITIONALCODE.
T.WRONG-UPDATE-STATE	Threat agent: Attacker Adverse action: An attacker prevents the OS Update operation to be performed atomically, resulting in an inconsistency between the resulting TOE code and the identification data: - The additional code is not loaded within the TOE, but the identification data is updated to mention that the additional code is present. - The additional code is loaded within the TOE, but the identification data is not updated to indicate the change. Directly threatened asset(s): D.OS-UPDATE-CODE-ID.
T.INTEG-OS-UPDATE-LOAD	Threat agent: Attacker Adverse action: The attacker modifies (part of) the additional code when it is transmitted to the TOE for installation. Directly threatened asset(s): D.OS-UPDATE_ADDITIONALCODE, D.JCS_CODE, D.JCS_DATA.
T.CONFID-OS-UPDATE-LOAD	Threat agent: Attacker Adverse action: The attacker discloses (part of) the additional code when it is transmitted to the TOE for installation. Directly threatened asset(s): D.OS-UPDATE_ADDITIONALCODE, D.JCS_CODE, D.JCS_DATA.

4.4 Organizational Security Policies

The definition of organizational security policies from [PP-eUICC] and [PP-JCS] is not repeated here. See section 3.4.2.4 for complete list is organizational security policies.

OSPs	
OSP.LIFE-CYCLE	
Additional OSP.OSP to cover OS-UPDATE	
OSP.ATOMIC_ACTIVATION	Additional code has to be loaded and installed on the Initial TOE through an atomic activation to create the Updated TOE. Each additional code shall be identified with unique Identification Data. During such atomic activation, identification Data of the Initial TOE have to be updated to clearly identify the Updated TOE. In case of interruption or incident during activation, the TOE shall remain in its initial state or fail secure.
OSP.TOE_IDENTIFICATION	Identification Data of the resulting Updated TOE shall identify the Initial TOE and the activated additional code. Identification Data shall be protected in integrity.
OSP.ADDITIONAL_CODE_SIGNING	The additional code has to be signed with a cryptographic key according to relevant standards, and the generated signature is associated with the additional code. The additional code signature must be verified during loading to assure its authenticity and integrity and to assure that loading is authorized on the TOE. The cryptographic key used to sign the additional code shall be of sufficient quality and its generation shall be appropriately secured to ensure the authenticity, integrity, and confidentiality of the key.
OSP.ADDITIONAL_CODE_ENCRYPTION	The additional code has to be encrypted according to the relevant standard in order to ensure its confidentiality when it is transmitted to the TOE for loading and installation. The encryption key shall be of sufficient quality and its generation shall be appropriately secured to ensure the confidentiality, authenticity, and integrity of the key.

4.5 Assumptions

Assumptions	
A.TRUSTED-PATHS-LPAd	
A.ACTORS	
A.APPLICATIONS	
Additional A.Assumption to cover OS-UPDATE	
A.OS-UPDATE-EVIDENCE	For additional code loaded pre-issuance, it is assumed that evaluated technical and/or audited organisational measures have been implemented to ensure that the additional code: 1. has been issued by the genuine OS Developer 2. has not been altered since it was issued by the genuine OS Developer. For additional code loaded post-issuance, it is assumed that the OS Developer provides digital evidence to the TOE in order to prove the following: 1. he is the genuine developer of the additional code and 2. The additional code has not been modified since it was issued by the genuine OS Developer.
A.SECURE_ACODE_ MANAGEMENT	It is assumed that: - The Key management process related to the OS Update capability takes place in a secure and audited environment. - The cryptographic keys used by the cryptographic operations are of strong quality and appropriately secured to ensure confidentiality, authenticity, and integrity of those keys.

5 SECURITY OBJECTIVES

This section introduces the security objectives for the TOE.

5.1 Security Objectives for the TOE

The list and definitions of the Security Objectives for the TOE from [PP-eUICC] are not repeated here. See section 3.4.3 for complete list is Security Objectives for the TOE.

Some objectives from the environment have been converted to objectives of the TOE, specifically the ones from [PP-eUICC] related to OE.RE* and OE.IC*. The replaced objectives from 3.4.3.2 and their description are listed next:

O.TOE	Replaced objectives description
O.PPE-PPI	
O.eUICC-DOMAIN-RIGHTS	
O.SECURE-CHANNELS	
O.INTERNAL-SECURE-CHANNELS	
O.PROOF_OF_IDENTITY	
O.OPERATE	
O.API	
O.DATA-CONFIDENTIALITY	
O.DATA-INTEGRITY	
O.ALGORITHMS	
O.IC.PROOF_OF IDENTITY	The underlying IC used by the TOE is uniquely identified.
O.IC.SUPPORT	The IC embedded software shall support the following functionalities: (1) It does not allow the TSFs to be bypassed or altered and does not allow access to low-level functions other than those made available by the packages of the API. That includes the protection of its private data and code (against disclosure or modification). (2) It provides secure low-level cryptographic processing to Profile Policy Enabler, Profile Package Interpreter, and Telecom Framework (S.PPE, S.PPI, and S.TELECOM). (3) It allows the S.PPE, S.PPI, and S.TELECOM to store data in "persistent technology memory" or in volatile memory, depending on its needs (for instance, transient objects must not be stored in non-volatile memory). The memory model is structured and allows for low-level control accesses (segmentation fault detection). (4) It provides a means to perform memory operations atomically for S.PPE, S.PPI, and S.TELECOM.
O.IC.RECOVERY	If there is a loss of power while an operation is in progress, the underlying IC must allow the TOE to eventually complete the interrupted operation

	successfully, or recover to a consistent and secure state.
O.RE.PPE-PPI	The Runtime Environment shall provide secure means for card management activities, including: o load of a package file, o installation of a package file, o extradition of a package file or an application, o personalization of an application or a Security Domain, o deletion of a package file or an application, o privileges update of an application or a Security Domain, o access to an application outside of its expected availability.
O.RE.SECURE-COMM	The Runtime Environment shall provide means to protect the confidentiality and integrity of applications communication.
O.RE.API	The Runtime Environment shall ensure that native code can be invoked only via an API.
O.RE.DATA-CONFIDENTIALITY	The Runtime Environment shall provide a means to protect at all times the confidentiality of the TOE sensitive data it processes.
O.RE.DATA-INTEGRITY	The Runtime Environment shall provide a means to protect at all times the integrity of the TOE sensitive data it processes.
O.RE.IDENTITY	The Runtime Environment shall ensure the secure identification of the applications it executes.
O.RE.CODE-EXE	The Runtime Environment shall prevent unauthorized code execution by applications.
Additional O.TOE to cover OS-UPDATE	
O.SECURE_LOAD_ACODE	The TOE shall check an evidence of authenticity and integrity of the additional code to be loaded. The TOE enforces that only an allowed version of the additional code can be loaded. The TOE shall forbid the loading of an additional code not intended to be assembled with the TOE. During the loading of the additional code, the TOE shall remain secure.
O.SECURE_AC_ACTIVATION	Activation of the additional code and update of the Identification Data shall be performed at the same time in an atomic way. All the operations needed for the code to be able to operate as in the Updated TOE shall be completed before activation. If the atomic activation is successful, then the resulting product is the Updated TOE, otherwise (in case of interruption or incident which prevents the forming of the Updated TOE), the TOE shall preserve a secure state.
O.TOE_IDENTIFICATION	The TOE provides means to store Identification Data in its non-volatile memory and guarantees the integrity of these data.

	After atomic activation of the additional code, the Identification Data of the Updated TOE allows identifications of both the Initial TOE and additional code. The user must be able to uniquely identify Initial TOE and additional code(s) which are embedded in the Updated TOE.
O.CONFID-OS-UPDATE.LOAD	The TOE shall decrypt the additional code prior installation. *Application Note*: Confidentiality protection must be enforced when the additional code is transmitted to the TOE for loading (See OE.OS-UPDATE-ENCRYPTION). Confidentiality protection can be achieved either through direct encryption of the additional code, or by means of a trusted path ensuring the confidentiality of the communication to the TOE.

5.2 Security Objectives for the Operational Environment

The list and definitions of the Security Objectives for the TOE from [PP-eUICC] are not repeated here. See section 3.4.3.2 for complete list is Security Objectives for the Operational Environment.

O.ENV	
OE.CI	A.ACTORS
OE.SM-DPplus	A.ACTORS
OE.MNO	A.ACTORS
OE.TRUSTED-PATHS-LPAd	A.TRUSTED-PATHS-LPAd
OE.APPLICATIONS	A.APPLICATIONS
OE.VERIFICATION	All the bytecodes shall be verified at least once, before the loading, before the installation or before the execution, depending on the card capabilities, in order to ensure that each bytecode is valid at execution time. See #.VERIFICATION for details. Additionally, the applet shall follow all the recommendations, if any, mandated in the platform guidance for maintaining the isolation property of the platform. Application Note: Constraints to maintain the isolation property of the platform are provided by the platform developer in application development guidance. The constraints apply to all application code loaded in the platform.
OE.CODE-EVIDENCE	A.APPLICATIONS
OE.MNO-SD	
Additional O.ENV to cover OS-UPDATE (from	
OE.OS-UPDATE-EVIDENCE	For additional code loaded pre issuance, evaluated technical measures implemented by the TOE or audited organisational measures must ensure that the additional code (1) has been issued by the genuine OS Developer and (2) has not been altered since it was issued by the genuine OS Developer. For additional code loaded post issuance, the OS Developer shall provide digital evidence to the TOE that (1) he is the genuine developer of the additional

	code and (2) the additional code has not been modified since it was issued by the genuine OS Developer.
OE.OS-UPDATE-ENCRYPTION	For additional code loaded post issuance, the OS Developer shall encrypt the additional code so that its confidentiality is ensured when it is transmitted to the TOE for loading and installation.
OE.SECURE_ACODE_MANAGEMENT	Key management processes related to the OS Update capability shall take place in a secure and audited environment. The key generation processes shall guarantee that cryptographic keys are of sufficient quality and appropriately secured to ensure confidentiality, authenticity, and integrity of the keys.

5.3 Security Objectives Rationale

5.3.1 Threats

5.3.1.1 Unauthorized profile and platform management

T.UNAUTHORIZED-PROFILE-MNG:

This threat is covered by requiring authentication and authorization from the legitimate actors:

- O.PPE-PPI and O.eUICC-DOMAIN-RIGHTS ensure that only authorized and authenticated actors (SM-DP+ and MNO OTA Platform) will access the Security Domains functions and content;
- OE.SM-DPplus and OE.MNO protect the corresponding credentials when used offcard. The on-card access control policy relies upon the underlying Runtime Environment, which ensures confidentiality and integrity of application data (O.RE.DATA-CONFIDENTIALITY and O.RE.DATA-INTEGRITY). The authentication is supported by corresponding secure channels:
- O.SECURE-CHANNELS and O.INTERNAL-SECURE-CHANNELS provide a secure channel for communication with SM-DP+ and a secure channel for communication with MNO OTA Platform. These secure channels rely upon the underlying Runtime Environment, which protects the applications communications (O.RE.SECURE-COMM).

Since the MNO-SD Security Domain is not part of the TOE, the operational environment has to guarantee that it will use securely the SCP80/81 secure channel provided by the TOE (OE.MNO-SD). In order to ensure the secure operation of the Application Firewall, the following objectives for the operational environment are also required:

- Compliance to security guidelines for applications (OE.APPLICATIONS, OE.VERIFICATION and OE.CODE-EVIDENCE).

T.UNAUTHORIZED-PLATFORM-MNG

This threat is covered by requiring authentication and authorization from the legitimate actors:

- O.PPE-PPI and O.eUICC-DOMAIN-RIGHTS ensure that only authorized and authenticated actors will access the Security Domains functions and content.

The on-card access control policy relies upon the underlying Runtime Environment, which ensures confidentiality and integrity of application data (O.RE.DATA-CONFIDENTIALITY and O.RE.DATA-INTEGRITY).

In order to ensure the secure operation of the Application Firewall, the following objectives for the operational environment are also required: o compliance to security guidelines for applications (OE.APPLICATIONS, OE.VERIFICATION and OE.CODE-EVIDENCE).

T.PROFILE-MNG-INTERCEPTION

Commands and profiles are transmitted by the SM-DP+ to its on-card representative (ISD-P), while profile data (including meta-data such as PPRs) is also transmitted by the MNO OTA Platform to its on-card representative (MNO-SD).

Consequently, the TSF ensures:

- Security of the transmission to the Security Domain (O.SECURE-CHANNELS and O.INTERNAL-SECURE-CHANNELS) by requiring authentication from SM-DP+ and MNO OTA Platforms, and protecting the transmission from unauthorized disclosure, modification and replay. These secure channels rely upon the underlying Runtime Environment, which protects the applications communications (O.RE.SECURE-COMM).

Since the MNO-SD Security Domain is not part of the TOE, the operational environment has to guarantee that it will securely use the SCP80/81 secure channel provided by the TOE (OE.MNO-SD). OE.SM-DPplus and OE.MNO ensure that the credentials related to the secure channels will not be disclosed when used by off-card actors.

T.PROFILE-MNG-ELIGIBILITY

Device Info and eUICCInfo2, transmitted by the eUICC to the SM-DP+, are used by the SM-DP+ to perform the Eligibility Check prior to allowing profile download onto the eUICC.

Consequently, the TSF ensures:

- Security of the transmission to the Security Domain (O.SECURE-CHANNELS and O.INTERNAL-SECURE-CHANNELS) by requiring authentication from SM-DP+, and protecting the transmission from unauthorized disclosure, modification and replay. These secure channels rely upon the underlying Runtime Environment, which protects the applications communications (O.RE.SECURE-COMM).

OE.SM-DPplus ensures that the credentials related to the secure channels will not be disclosed when used by off-card actors. O.DATA-INTEGRITY and O.RE.DATA-INTEGRITY ensure that the integrity of Device Info and eUICCInfo2 is protected at the eUICC level.

5.3.1.2 Identity Tampering

T.UNAUTHORIZED-IDENTITY-MNG

O.PPE-PPI and O.eUICC-DOMAIN-RIGHTS covers this threat by providing an access control policy for ECASD content and functionality.

The on-card access control policy relies upon the underlying Runtime Environment, which ensures confidentiality and integrity of application data (O.RE.DATA-CONFIDENTIALITY and O.RE.DATA-INTEGRITY).

O.RE.IDENTITY ensures that at the Java Card level, the applications cannot impersonate other actors or modify their privileges.

T.IDENTITY-INTERCEPTION

O.INTERNAL-SECURE-CHANNELS ensures the secure transmission of the shared secrets from the ECASD to ISD-R and ISD-P. These secure channels rely upon the underlying Runtime Environment, which protects the applications communications (O.RE.SECURE-COMM).

OE.CI ensures that the CI root will manage securely its credentials off-card.

5.3.1.3 eUICC cloning

T.UNAUTHORIZED-eUICC

O.PROOF_OF_IDENTITY guarantees that the off-card actor can be provided with a cryptographic proof of identity based on an EID.

O.PROOF_OF_IDENTITY guarantees this EID uniqueness by basing it on the eUICC hardware identification (which is unique due to O.IC.PROOF_OF_IDENTITY).

5.3.1.4 LPAd impersonation

T.LPAd-INTERFACE-EXPLOIT

OE.TRUSTED-PATHS-LPAd ensures that the interfaces ES10a, ES10b and ES10c are trusted paths to the LPAd.

5.3.1.5 Unauthorized access to the mobile network

T.UNAUTHORIZED-MOBILE-ACCESS

The objective O.ALGORITHMS ensures that a profile may only access the mobile network using a secure authentication method, which prevents impersonation by an attacker.

5.3.1.6 Second Level Threats

T.LOGICAL-ATTACK

This threat is covered by controlling the information flow between Security Domains and the PPE, PPI, the Telecom Framework or any native/OS part of the TOE. As such it is covered:

- by the APIs provided by the Runtime Environment (O.RE.API);
- by the APIs of the TSF (O.API); the APIs of Telecom Framework, PPE and PPI shall ensure atomic transactions (O.IC.SUPPORT).

Whenever sensitive data of the TOE are processed by applications, confidentiality and integrity must be protected at all times by the Runtime Environment (O.RE.DATACONFIDENTIALITY, O.RE.DATA-INTEGRITY). However these sensitive data are also processed by the PPE, PPI and the Telecom Framework, which are not protected by these mechanisms. Consequently,

- the TOE itself must ensure the correct operation of PPE, PPI and Telecom Framework (O.OPERATE), and
- PPE, PPI and Telecom Framework must protect the confidentiality and integrity of the sensitive data they process, while applications must use the protection mechanisms provided by the Runtime Environment (O.DATA-CONFIDENTIALITY, O.DATA-INTEGRITY).

The following objectives for the operational environment are also required:

- prevention of unauthorized code execution by applications (O.RE.CODE-EXE),
- Compliance to security guidelines for applications (OE.APPLICATIONS, OE.VERIFICATION and OE.CODE-EVIDENCE).

T.PHYSICAL-ATTACK

This threat is countered mainly by physical protections which rely on the underlying Platform and are therefore an environmental issue.

The security objectives O.IC.SUPPORT and O.IC.RECOVERY protect sensitive assets of the Platform against loss of integrity and confidentiality and especially ensure the TSFs cannot be bypassed or altered.

In particular, the security objective O.IC.SUPPORT provides functionality to ensure atomicity of sensitive operations, secure low level access control and protection against bypassing of the security features of the TOE. In particular, it explicitly ensures the independent protection in integrity of the Platform data.

Since the TOE cannot only rely on the IC protection measures, the TOE shall enforce any necessary mechanism to ensure resistance against side channels (O.DATACONFIDENTIALITY). For the same reason, the Java Card Platform security architecture must cover side channels (O.RE.DATA-CONFIDENTIALITY).

5.3.2 Organizational Security Policies

The OSP defined is OSP.LIFE-CYCLE as in [PP-eUICC].

OSPs
OSP.LIFE-CYCLE

The OSP defined is OSP.ATOMIC_ACTIVATION, OSP.TOE_IDENTIFICATION, OSP.ADDITIONAL_CODE_SIGNING, and OSP.ADDITIONAL_CODE_ENCRYPTION as in [PP-GP].

OSPs
OSP.ATOMIC_ACTIVATION
OSP.TOE_IDENTIFICATION
OSP.ADDITIONAL_CODE_SIGNING
OSP.ADDITIONAL_CODE_ENCRYPTION

5.3.3 Assumptions

The assumptions A.TRUSTED-PATHS-LPAd, A.ACTORS and A.APPLICATIONS are defined as in [PP-eUICC].

Assumptions
A.TRUSTED-PATHS-LPAd
A.ACTORS
A.APPLICATIONS

The Assumption defined is A.OS-UPDATE-EVIDENCE, A.SECURE_ACODE_MANAGEMENT as in [PP-GP].

OSPs
A.OS-UPDATE-EVIDENCE
A.SECURE_ACODE_MANAGEMENT

5.3.4 Rationale tables

5.3.4.1 Threats Rationale

Threats	Security Objectives	Rationale
T.UNAUTHORIZEDPROFILE-MNG	O.eUICC-DOMAIN-RIGHTS, OE.SM-DPplus, OE.MNO, O.PPE-PPI, O.SECURE-CHANNELS, OE.APPLICATIONS,OE.VERIFICATION OE.CODE-EVIDENCE, O.INTERNAL-SECURECHANNELS, O.RE.SECURE-COMM, O.RE.DATACONFIDENTIALITY, O.RE.DATA-INTEGRITY, OE.MNO-SD	Section 5.3.1.1
T.UNAUTHORIZEDPLATFORM-MNG	O.eUICC-DOMAIN-RIGHTS, O.PPE-PPI, OE.APPLICATIONS,OE.VERIFICATION OE.CODE-EVIDENCE, O.RE.DATA-CONFIDENTIALITY, O.RE.DATA-INTEGRITY	Section 5.3.1.1
T.PROFILE-MNG-INTERCEPTION	OE.SM-DPplus, OE.MNO, O.SECURE-CHANNELS, O.INTERNAL-SECURE-CHANNELS,O.RE.SECURE-COMM, OE.MNO-SD	Section 5.3.1.1

T.PROFILE-MNG-ELIGIBILITY	OE.SM-DPplus, O.RE.SECURE-COMM, O.SECURE-CHANNELS, O.INTERNAL-SECURE-CHANNELS, O.RE.DATA-INTEGRITY, O.DATA-INTEGRITY	Section 5.3.1.1
T.UNAUTHORIZED-IDENTITY-MNG	O.eUICC-DOMAIN-RIGHTS, O.PPE-PPI, O.RE.DATA-CONFIDENTIALITY, O.RE.DATA-INTEGRITY, O.RE.IDENTITY	Section 5.3.1.2
T.IDENTITY-INTERCEPTION	OE.CI, O.INTERNAL-SECURE-CHANNELS, O.RE.SECURE-COMM	Section 5.3.1.2
T.UNAUTHORIZED-eUICC	O.PROOF_OF_IDENTITY, O.IC.PROOF_OF_IDENTITY	Section 5.3.1.3
T.LPAd-INTERFACE-EXPLOIT	OE.TRUSTED-PATHS-LPAd	Section 5.3.1.4
T.UNAUTHORIZED-MOBILE-ACCESS	O.ALGORITHMS	Section 5.3.1.5
Threats	**Security Objectives**	**Rationale**
T.LOGICAL-ATTACK	O.DATA-CONFIDENTIALITY, O.DATA-INTEGRITY, O.API, OE.APPLICATIONS , OE.VERIFICATION, OE.CODE-EVIDENCE**,** O.OPERATE, O.RE.API, O.RE.CODE-EXE, O.IC.SUPPORT, O.RE.DATA-CONFIDENTIALITY, O.RE.DATA-INTEGRITY	Section 5.3.1.6
T.PHYSICAL-ATTACK	O.IC.SUPPORT, O.IC.RECOVERY, O.DATA-CONFIDENTIALITY, O.RE.DATA-CONFIDENTIALITY	Section 5.3.1.6
T.UNAUTHORISED-TOE-CODE-UPDATE	O.SECURE_LOAD_ACODE	Section 5.1
T.FAKE-SGNVER-KEY	O.SECURE_LOAD_ACODE	Section 5.1
T.WRONG-UPDATE-STATE	O.SECURE_AC_ACTIVATION, O.TOE_IDENTIFICATION	Section 5.1
T.INTEG-OS-UPDATE-LOAD	O.SECURE_LOAD_ACODE	Section 5.1
T.CONFID-OS-UPDATE-LOAD	O.CONFID-OS-UPDATE.LOAD	Section 5.1

Table 13 - Threats and Security Objectives- Coverage

Security Objectives	**Threats**
O.PPE-PPI	T.UNAUTHORIZED-PROFILE-MNG, T.UNAUTHORIZED-PLATFORM-MNG, T.UNAUTHORIZED-IDENTITY-MNG
O.eUICC-DOMAIN-RIGHTS	T.UNAUTHORIZED-PROFILE-MNG, T.UNAUTHORIZED-PLATFORM-MNG, T.UNAUTHORIZED-IDENTITY-MNG
O.SECURE-CHANNELS	T.UNAUTHORIZED-PROFILE-MNG, T.PROFILE-MNG-INTERCEPTION, T.PROFILE-MNG-ELIGIBILITY
O.INTERNAL-SECURE-CHANNELS	T.UNAUTHORIZED-PROFILE-MNG, T.PROFILE-MNG-INTERCEPTION, T.PROFILE-MNG-ELIGIBILITY, T.IDENTITY-INTERCEPTION
O.PROOF_OF_IDENTITY	T.UNAUTHORIZED-eUICC
O.OPERATE	T.LOGICAL-ATTACK
O.API	T.LOGICAL-ATTACK
O.DATA-CONFIDENTIALITY	T.LOGICAL-ATTACK, T.PHYSICAL-ATTACK
O.DATA-INTEGRITY	T.PROFILE-MNG-ELIGIBILITY, T.LOGICAL-ATTACK

O.ALGORITHMS	T.UNAUTHORIZED-MOBILE-ACCESS
OE.CI	T.IDENTITY-INTERCEPTION
OE.SM-DPplus	T.UNAUTHORIZED-PROFILE-MNG, T.PROFILE-MNG-INTERCEPTION, T.PROFILE-MNG-ELIGIBILITY
OE.MNO	T.UNAUTHORIZED-PROFILE-MNG, T.PROFILE-MNG-INTERCEPTION
O.IC.PROOF_OF_IDENTITY	T.UNAUTHORIZED-eUICC
O.IC.SUPPORT	T.LOGICAL-ATTACK, T.PHYSICAL-ATTACK
O.IC.RECOVERY	T.PHYSICAL-ATTACK
O.RE.PPE-PPI	
O.RE.SECURE-COMM	T.UNAUTHORIZED-PROFILE-MNG, T.PROFILE-MNG-INTERCEPTION, T.PROFILE-MNG-ELIGIBILITY, T.IDENTITY-INTERCEPTION
O.RE.API	T.LOGICAL-ATTACK
O.RE.DATA-CONFIDENTIALITY	T.UNAUTHORIZED-PROFILE-MNG, T.UNAUTHORIZED-PLATFORM-MNG, T.UNAUTHORIZED-IDENTITY-MNG, T.LOGICAL-ATTACK, T.PHYSICAL-ATTACK
O.RE.DATA-INTEGRITY	T.UNAUTHORIZED-PROFILE-MNG, T.UNAUTHORIZED-PLATFORM-MNG, T.PROFILE-MNG-ELIGIBILITY, T.UNAUTHORIZED-IDENTITY-MNG, T.LOGICAL-ATTACK
O.RE.IDENTITY	T.UNAUTHORIZED-IDENTITY-MNG
O.RE.CODE-EXE	T.LOGICAL-ATTACK
OE.TRUSTED-PATHS-LPAd	T.LPAd-INTERFACE-EXPLOIT
OE.APPLICATIONS	T.UNAUTHORIZED-PROFILE-MNG, T.UNAUTHORIZED-PLATFORM-MNG, T.LOGICAL-ATTACK
OE.VERIFICATION	T.UNAUTHORIZED-PROFILE-MNG, T.UNAUTHORIZED-PLATFORM-MNG, T.LOGICAL-ATTACK
OE.CODE-EVIDENCE	T.UNAUTHORIZED-PROFILE-MNG, T.UNAUTHORIZED-PLATFORM-MNG, T.LOGICAL-ATTACK
OE.MNO-SD	T.UNAUTHORIZED-PROFILE-MNG, T.PROFILE-MNG-INTERCEPTION

Table 14 - Security Objectives and threats

5.3.4.2 Organizational Security Policies Rationale

Organizational Security Policies	Security Objectives	Rationale
OSP.LIFE-CYCLE	O.PPE-PPI, O.RE.PPE-PPI, O.OPERATE	Section 5.1
OSP.ATOMIC_ACTIVATION	O.SECURE_AC_ACTIVATION	Section 5.3.2
OSP.TOE_IDENTIFICATION	O.TOE_IDENTIFICATION	Section 5.3.2
OSP.ADDITIONAL_CODE_SIGNING	O.SECURE_LOAD_ACODE	Section 5.3.2
OSP.ADDITIONAL_CODE_ENCRYPTION	O.CONFID-OS-UPDATE.LOAD, OE.OS-UPDATE-ENCRYPTION	Section 5.3.2

Table 15 - Organizational Security Policies and Security Objectives- Coverage

Security Objectives	Organizational Security Policies
O.PPE-PPI	OSP.LIFE-CYCLE
O.eUICC-DOMAIN-RIGHTS	
O.SECURE-CHANNELS	
O.INTERNAL-SECURE-CHANNELS	

O.PROOF_OF_IDENTITY	
O.OPERATE	OSP.LIFE-CYCLE
O.API	
O.DATA-CONFIDENTIALITY	
O.DATA-INTEGRITY	
O.ALGORITHMS	
OE.CI	
OE.SM-DPplus	
OE.MNO	
O.IC.PROOF_OF_IDENTITY	
O.IC.SUPPORT	
O.IC.RECOVERY	
O.RE.PPE-PPI	OSP.LIFE-CYCLE
O.RE.SECURE-COMM	
O.RE.API	
O.RE.DATA-CONFIDENTIALITY	
O.RE.DATA-INTEGRITY	
O.RE.IDENTITY	
O.RE.CODE-EXE	
OE.TRUSTED-PATHS-LPAd	
OE.APPLICATIONS	
OE.VERIFICATION	
OE.CODE-EVIDENCE	
OE.MNO-SD	
OE.SM-DS	

Table 16 - Security Objectives and Organizational Security Policies

5.3.4.3 Assumptions Rationale

Assumptions	**Security Objectives for the Operational Environment**	**Rationale**
A.TRUSTED-PATHS-LPAd	OE.TRUSTED-PATHS-LPAd	Section 5.3.3
A.ACTORS	OE.CI, OE.SM-DPplus, OE.MNO	Section 5.3.3
A.APPLICATIONS	OE.APPLICATIONS,OE.VERIFICATION OE.CODE-EVIDENCE	Section 5.3.3
A.OS-UPDATE-EVIDENCE	OE.OS-UPDATE-EVIDENCE	Section 5.3.3
A.SECURE_ACODE_MANAGEMENT	OE.SECURE_ACODE_MANAGEMENT	Section 5.3.3

Table 17 - Assumptions and Security Objectives for the Operational Environment- Coverage

Security Objectives for the Operational Environment	**Assumptions**
OE.CI	A.ACTORS
OE.SM-DPplus	A.ACTORS
OE.MNO	A.ACTORS
OE.IC.PROOF_OF_IDENTITY	
OE.IC.SUPPORT	
OE.IC.RECOVERY	
OE.RE.PPE-PPI	
OE.RE.SECURE-COMM	
OE.RE.API	

OE.RE.DATA-CONFIDENTIALITY	
OE.RE.DATA-INTEGRITY	
OE.RE.IDENTITY	
OE.RE.CODE-EXE	
OE.TRUSTED-PATHS-LPAd	A.TRUSTED-PATHS-LPAd
OE.APPLICATIONS	A.APPLICATIONS
OE.VERIFICATION	A.APPLICATIONS
OE.CODE-EVIDENCE	A.APPLICATIONS
OE.MNO-SD	

Table 18 - Assumptions and Security Objectives for the Operational Environment

6 EXTENDED COMPONENTS DEFINITION

The same extended component definition than [PP-eUICC] and [PP-84] are defined in the current Security target:

- Extended Family FIA_API - Authentication Proof of Identity
- Extended Family FPT_EMS - TOE Emanation
- Extended Family FCS_RNG – Random number generation
- Extended Family FAU_SAS – Audit Data Storage

The extended components definition (FIA_API, FPT_EMS and FCS_RNG) from [PP-eUICC] is not repeated here. The same for FAU_SAS.1 which definition from [PP-84], section 5.3 have been taken with no modification.

7 SECURITY REQUIREMENTS

For section 7.1, the following conventions are used in the definitions of the SFRs:

- Selections and assignments that have already been made in the [PP-eUICC] are in **bold**, and the original text on which the selection or assignment has been made is not reminded.
- Selections and assignments made in this ST are in blue or **bold blue**.

- This convention also applies for section 7.2 for SFRs in light blue (as example below) dedicated to RSP module.

- **example**

Iteration operations on SFR components are denoted by showing a slash "/" and the iteration indicator after the SFR component identifier. For section 7.2, following conventions are used in the definitions of the SFRs :

- Selections and assignments that have already been made in the [PP-GP] and [PP-JCS] Protection Profiles are **in bold**, and the original text on which the selection or assignment has been made is not reminded.
- Selections and assignments made in this ST are **in bold and underlined**, and the PP original text on which the selection or assignment has been made is indicated in a footnote.
- Iteration operations on SFR components are denoted by showing a slash "/" and the iteration indicator after the SFR component identifier.

- This convention applies for SFRs in dark blue as example below:

- **example**

7.1 eUICC Security Functional Requirements

The introduction and security attributes definition are present in [PP-eUICC] section 6.1 and are not repeated here.

7.1.1 Identification and authentication

FIA_UID.1/EXT Timing of identification

FIA_UID.1.1/EXT The TSF shall allow

- **application selection**
- **requesting data that identifies the eUICC**
- **[assignment: list of none].**

On behalf of the user to be performed before the user is identified.

FIA_UID.1.2/EXT The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.1/EXT Timing of authentication

FIA_UAU.1.1/EXT The TSF shall allow

- **application selection**
- **requesting data that identifies the eUICC**
- **user identification**
- **[assignment: none]**

On behalf of the user to be performed before the user is authenticated.

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

FIA_USB.1/EXT User-subject binding

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

- **SM-DP+ OID is associated to S.ISD-R, acting on behalf of U.SM-DPplus**
- **MNO OID is associated to U.MNO-SD, acting on behalf of U.MNO-OTA**.

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

- **Initial association of SM-DP+ OID and MNO OID requires U.SM-DPplus to be authenticated via "CERT.DPauth.ECDSA"**.

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

- **change of SM-DP+ OID requires U.SM-DPplus to be authenticated via "CERT.DPauth.ECDSA"**
- **change of MNO OID is not allowed**.

FIA_UAU.4/EXT Single-use authentication mechanisms

FIA_UAU.4.1/EXT The TSF shall prevent reuse of authentication data related to **the authentication mechanism used to open a secure communication channel between the eUICC and**

- **U.SM-DPplus**
- **U.MNO-OTA**.

FIA_UID.1/MNO-SD Timing of identification

FIA_UID.1.1/MNO-SD The TSF shall allow [assignment: **none**] on behalf of the user to be performed before the user is identified.

FIA_UID.1.2/MNO-SD The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_USB.1/MNO-SD User-subject binding

FIA_USB.1.1/MNO-SD The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: **The U.MNO-SD AID is associated to the S.ISD-P acting on behalf of U.MNO-SD**.

FIA_USB.1.2/MNO-SD The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: **Initial association of AID requires U.SM-DP+ to be authenticated via CERT.DPauth.ECDSA**.

FIA_USB.1.3/MNO-SD The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: **no change of AID is allowed**.

FIA_ATD.1 User attribute definition

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users:

- **CERT.DPauth.ECDSA, CERT.DPpb.ECDSA, and SM-DP+ OID belonging to U.SM-DPplus;**
- **MNO OID belonging to U.MNO-OTA;**
- **AID belonging to U.MNO-SD**.

FIA_API.1 Authentication Proof of Identity

FIA_API.1.1 The TSF shall provide a **cryptographic authentication mechanism based on the EID of the eUICC** to prove the identity of the TOE to an external entity.

7.1.2 Communication

FDP_IFC.1/SCP Subset information flow control

FDP_IFC.1.1/SCP The TSF shall enforce the **Secure Channel Protocol information flow control SFP** on

- **users/subjects:**
  - **U.SM-DPplus and S.ISD-R**
  - **U.MNO-OTA and U.MNO-SD**
- **information: transmission of commands.**

FDP_IFF.1/SCP Simple security attributes

FDP_IFF.1.1/SCP The TSF shall enforce the **Secure Channel Protocol information flow control SFP** based on the following types of subject and information security attributes:

- **users/subjects:**
  - **U.SM-DPplus and S.ISD-R, with security attribute D.SECRETS**
  - **U.MNO-OTA and U.MNO-SD, with security attribute D.MNO_KEYS**
- **information: transmission of commands**.

FDP_IFF.1.2/SCP The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:

- **The TOE shall permit communication between U.MNO-OTA and U.MNOSD in a SCP80 or SCP81 secure channel**.

FDP_IFF.1.3/SCP The TSF shall enforce the [assignment: **none**].

FDP_IFF.1.4/SCP The TSF shall explicitly authorise an information flow based on the following rules: [assignment: **none**].

FDP_IFF.1.5/SCP The TSF shall explicitly deny an information flow based on the following rules:

- **The TOE shall reject communication between U.SM-DPplus and S.ISD-R if it is not performed in a SCP-SGP22 secure channel**.

FTP_ITC.1/SCP Inter-TSF trusted channel

FTP_ITC.1.1/SCP The TSF shall provide a communication channel between itself and another trusted IT product 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/SCP The TSF shall permit another trusted IT product to initiate communication via the trusted channel.

FTP_ITC.1.3/SCP The TSF shall initiate communication via the trusted channel for [assignment: **following list of functions for which a trusted channel is required**].

The TSF shall permit the SM-DP+ to open a SCP-SGP22 secure channel to transmit the following operations:

- ES8+.InitialiseSecureChannel

- ES8+.ConfigureISDP
- ES8+.StoreMetadata
- ES8+.ReplaceSessionKeys
- ES8+.LoadProfileElements.

The TSF shall permit the LPAd to transmit the following operations:

- ES10a.GetEuiccConfiguredAddresses
- ES10a.SetDefaultDpAddress
- ES10b.PrepareDownload
- ES10b.LoadBoundProfilePackage
- ES10b.GetEUICCChallenge
- ES10b.GetEUICCInfo
- ES10b.ListNotification
- ES10b.RetrieveNotificationsList
- ES10b.RemoveNotificationFromList
- ES10b.AuthenticateServer
- ES10b.CancelSession
- ES10c.GetProfilesInfo
- ES10c.EnableProfile
- ES10c.DisableProfile
- ES10c.DeleteProfile
- ES10c.eUICCMemoryReset
- ES10c.GetEID
- ES10c.SetNickname
- ES10c.GetRAT.

The TSF shall permit the remote OTA Platform to open a SCP80 or SCP81 secure channel to transmit the following operation:

- ES6.UpdateMetadata.

FDP_ITC.2/SCP Import of user data with security attributes

FDP_ITC.2.1/SCP The TSF shall enforce the **Secure Channel Protocol information flow control SFP** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/SCP The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/SCP The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

FDP_ITC.2.4/SCP The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5/SCP The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: [assignment: **none**].

FPT_TDC.1/SCP Inter-TSF basic TSF data consistency

FPT_TDC.1.1/SCP The TSF shall provide the capability to consistently interpret

- **Commands from U.SM-DPplus and U.MNO-OTA**
- **Downloaded objects from U.SM-DPplus and U.MNO-OTA**

When shared between the TSF and another trusted IT product.

FPT_TDC.1.2/SCP The TSF shall use [assignment: **none**] when interpreting the TSF data from another trusted IT product.

FDP_UCT.1/SCP Basic data exchange confidentiality

FDP_UCT.1.1/SCP The TSF shall enforce the **Secure Channel Protocol information flow control SFP** to receive user data in a manner protected from unauthorised disclosure.**FDP_UIT.1/SCP Data exchange integrity**

FDP_UIT.1.1/SCP The TSF shall enforce the **Secure Channel Protocol information flow control SFP** to receive user data in a manner protected from modification, deletion, insertion and replay errors.

FDP_UIT.1.2/SCP The TSF shall be able to determine on receipt of user data, whether modification, deletion, insertion and replay has occurred.

FCS_CKM.1/SCP-SM Cryptographic key generation

FCS_CKM.1.1/SCP-SM The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **ElGamal elliptic curves key agreement (ECKA)** and specified cryptographic key sizes **256** that meet the following: **ECKA-EG using one of the following standards:**

- **NIST P-256 (FIPS PUB 186-3 Digital Signature Standard)**
- **brainpoolP256r1 (BSI TR-03111, Version 1.11, RFC 5639)**
- **FRP256V1 (ANSSI ECC FRP256V1)**.

Note: in this TOE, the **FRP256V1 (ANSSI ECC FRP256V1)** is not supported

FCS_CKM.2/SCP-MNO Cryptographic key distribution

FCS_CKM.2.1/SCP-MNO The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: **distribution method from SCP-SGP22 (SCP03t)**] that meets the following: [assignment: **SGP.22 standard**].

FCS_CKM.4/SCP-SM Cryptographic key destruction

FCS_CKM.4.1/SCP-SM The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: **wipe the buffer with random bytes**] that meets the following: [assignment: **none**].

FCS_CKM.4/SCP-MNO Cryptographic key destruction

FCS_CKM.4.1/SCP-MNO The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: **clearKey() method**] that meets the following: [assignment: **[JCAPI3]**].

7.1.3 Security Domains

FDP_ACC.1/ISDR Subset access control

FDP_ACC.1.1/ISDR The TSF shall enforce the **ISD-R access control SFP** on

- **subjects: S.ISD-R**
- **objects: S.ISD-P**
- **operations:**
  - **Create and configure profile**
  - **Store profile metadata**
  - **Enable profile**
  - **Disable profile**
  - **Delete profile**
  - **Perform a Memory reset**.

FDP_ACF.1/ISDR Security attribute based access control

FDP_ACF.1.1/ISDR The TSF shall enforce the **ISD-R access control SFP** to objects based on the following:

- **subjects: S.ISD-R**
- **objects:**
  - **S.ISD-P with security attributes "state" and "PPR"**
- **operations:**
  - **Create and configure profile**
  - **Store profile metadata**

- **Enable profile**
- **Disable profile**
- **Delete profile**
- **Perform a Memory reset**.

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

- **Enabling a S.ISD-P is authorized only if**
  - **the corresponding S.ISD-P is in the state "DISABLED" and**
  - **the currently enabled S.ISD-P's PPR data allows its disabling.**
- **Disabling a S.ISD-P is authorized only if**
  - **the corresponding S.ISD-P is in the state "ENABLED" and**
  - **the corresponding S.ISD-P's PPR data allows its disabling.**
- **Deleting a S.ISD-P is authorized only if**
  - **the corresponding S.ISD-P is not in the state "ENABLED" and**
  - **the corresponding S.ISD-P's PPR data allows its deletion.**
- **Performing a S.ISD-P Memory reset is authorized regardless of the involved S.ISD-P's state or PPR attribute**.

FDP_ACF.1.3/ISDR The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: **none**].

FDP_ACF.1.4/ISDR The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: **none**].

FDP_ACC.1/ECASD Subset access control

FDP_ACC.1.1/ECASD The TSF shall enforce the **ECASD access control SFP** on

- **subjects: S.ISD-R,**
  objects: S.ECASD,
  operations:
  - **execution of a ECASD function**
  - **access to output data of these functions,**
- **[assignment: none].**

FDP_ACF.1/ECASD Security attribute based access control

FDP_ACF.1.1/ECASD The TSF shall enforce the **ECASD access control SFP** to objects based on the following:

- **subjects: S.ISD-R, with security attribute "AID"**
  objects: S.ECASD
  operations:
  - **execution of a ECASD function**
    - **Verification of the off-card entities Certificates (SM-DP+, SM-DS), provided by an ISD-R, with the CI public key (PK.CI.ECDSA)**

        - **Creation of a eUICC signature on material provided by an ISD-R.**
    - **access to output data of these functions**.
- **[assignment: none].**

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

- **Authorized users: only S.ISD-R, identified by its AID, shall be authorized to execute the following S.ECASD functions:**
    - **Verification of a certificate CERT.DPauth.ECDSA, CERT.DPpb.ECDSA, CERT.DP.TLS, CERT.DSauth.ECDSA, or CERT.DS.TLS, provided by an ISD-R, with the CI public key (PK.CI.ECDSA)**
    - **Creation of a eUICC signature, using D.SK.EUICC.ECDSA, on material provided by an ISD-R**.
- **[assignment: none].**

FDP_ACF.1.3/ECASD The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: **none**].

FDP_ACF.1.4/ECASD The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: **none**].

7.1.4 Platform Services

FDP_IFC.1/Platform_services Subset information flow control

FDP_IFC.1.1/Platform_services The TSF shall enforce the **Platform services information flow control SFP** on
users/subjects:

- **S.ISD-R, S.ISD-P, U.MNO-SD**
- **Platform code (S.PPE, S.PPI, S.TELECOM)**

information:

- **D.PROFILE_NAA_PARAMS**
- **D.PROFILE_POLICY_RULES**
- **D.PLATFORM_RAT**

operations:

- **installation of a profile**
- **PPR and RAT enforcement**
- **network authentication.**

FDP_IFF.1/Platform_services Simple security attributes

FDP_IFF.1.1/Platform_services The TSF shall enforce the **Platform services information flow control SFP** based on the following types of subject and information security attributes:

users/subjects:

- **S.ISD-R, S.ISD-P, U.MNO-SD, with security attribute "application identifier (AID)"**

information:

- **D.PROFILE_NAA_PARAMS**
- **D.PROFILE_POLICY_RULES**
- **D.PLATFORM_RAT**

operations:

- **installation of a profile**
- **PPR and RAT enforcement**
- **network authentication.**

FDP_IFF.1.2/Platform_services The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:

- **D.PROFILE_NAA_PARAMS shall be transmitted only:**
  - **by U.MNO-SD to S.TELECOM in order to execute the network authentication function**
  - **by S.ISD-R to S.PPI using the profile installation function**
- **D.PROFILE_POLICY_RULES shall be transmitted only**
  - **by S.ISD-R to S.PPE in order to execute the PPR enforcement function** o

D.PLATFORM_RAT shall be transmitted only

- **by S.ISD-R to S.PPE in order to execute the RAT enforcement function**.

FDP_IFF.1.3/Platform_services The TSF shall enforce the [assignment: **none**].

FDP_IFF.1.4/Platform_services The TSF shall explicitly authorise an information flow based on the following rules: [assignment: **none**].

FDP_IFF.1.5/Platform_services The TSF shall explicitly deny an information flow based on the following rules: [assignment: **none**].

FPT_FLS.1/Platform_services Failure with preservation of secure state

FPT_FLS.1.1/Platform_services The TSF shall preserve a secure state when the following types of failures occur:

- **failure that lead to a potential security violation during the processing of a S.PPE, S.PPI or S.TELECOM API specific functions:**
  - **Installation of a profile**
  - **PPR and RAT enforcement**
  - **Network authentication**
- **[assignment: none]**.

7.1.5 Security management

FCS_RNG.1 Random Number Generation

FCS_RNG.1.1 The TSF shall provide a [selection: **hybrid deterministic**] random number generator [selection: **DRG.4**] that implements: [assignment: **Hybrid design, Forward secrecy, Enhanced backward secrecy, Enhanced forward secrecy, Entropy input quality**].

Application Note:

- Hybrid design: *(DRG.4.1) "The internal state of the RNG shall use PTRNG of class PTG.2 as random source".*
- Forward secrecy: *(DRG.4.2) "The RNG provides forward secrecy".*
- Enhanced backward secrecy: *(DRG.4.3) "The RNG provides backward secrecy even if the current internal state is known".*
- Enhanced forward secrecy: *(DRG.4.4) "The RNG provides enhanced forward secrecy after calling the JAVA API "ALG_KEYGENERATION" or "ALG_TRNG"".*
- Entropy input quality: *(DRG.4.5) "The internal state of the RNG is seeded by an PTRNG of class PTG.2".*

FCS_RNG.1.2 The TSF shall provide random numbers that meet [assignment: **Output and mutual difference**, **Statistical tests**].

Application Note:

- Output and mutual difference: *(DRG.4.6) "The RNG generates output for which* 2^{35} *strings of bit length 128 are mutually different with probability greater than or equal to* $1 - \frac{1}{2^{58}}$ *".*
- Statistical tests: *(DRG.4.7) "Statistical tests suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A [AIS31]".*

FPT_EMS.1 TOE Emanation

FPT_EMS.1.1 The TOE shall not emit [assignment: **side channels (power consumptions and electromagnetic fluctuations)**] in excess of [assignment: **IC limits**] enabling access to

- **D.SECRETS;**
- **D.SK.EUICC.ECDSA**

and **the secret keys which are part of the following keysets:**

- **D.MNO_KEYS,**
- **D.PROFILE_NAA_PARAMS.**

FPT_EMS.1.2 The TSF shall ensure [assignment: **users**] are unable to use the following interface [assignment: **secure processor communication**] to gain access to

- **D.SECRETS;**
- **D.SK.EUICC.ECDSA**

and **the secret keys which are part of the following keysets:**

- **D.MNO_KEYS,**
- **D.PROFILE_NAA_PARAMS.**

FDP_SDI.1 Stored data integrity monitoring

FDP_SDI.1.1 The TSF shall monitor user data stored in containers controlled by the TSF for **integrity errors** on all objects, based on the following attributes: **integrity-sensitive data**.

Refinement:
The notion of integrity-sensitive data covers the assets of the Security Target TOE that require to be protected against unauthorized modification, including but not limited to the assets of this PP that require to be protected against unauthorized modification:

o D.MNO_KEYS

o Profile data

- D.PROFILE_NAA_PARAMS
- D.PROFILE_IDENTITY
- D.PROFILE_POLICY_RULES
- D.PROFILE_USER_CODES

o Management data

- D.PLATFORM_DATA
- D.DEVICE_INFO
- D.PLATFORM_RAT

o Identity management data

- D.SK.EUICC.ECDSA
- D.CERT.EUICC.ECDSA
- D.PK.CI.ECDSA
- D.EID
- D.SECRETS
- D.CERT.EUM.ECDSA
- D.CRLs if existing

FDP_RIP.1 Subset residual information protection

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource from and allocation of the resource to the following objects:

- **D.SECRETS;**
- **D.SK.EUICC.ECDSA;**
- **The secret keys which are part of the following keysets:**
  - **D.MNO_KEYS,**
  - **D.PROFILE_NAA_PARAMS**.

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:

- **failure of creation of a new ISD-P by ISD-R**
- **failure of installation of a profile by ISD-R.**

FMT_MSA.1/PLATFORM_DATA Management of security attributes

FMT_MSA.1.1/PLATFORM_DATA The TSF shall enforce the **ISD-R access control policy** to restrict the ability to modify the security attributes **the following parts of D.PLATFORM_DATA:**

- **ISD-P state**

to

- **S.ISD-R to modify ISD-P state**
  - **from "INSTALLED" to "SELECTABLE" (during ISD-P creation)**
  - **from "ENABLED" to "DISABLED" (during profile disabling)**
- **S.ISD-R to modify ISD-P state**
  - **from "DISABLED" to "ENABLED" (during profile enabling)**.

FMT_MSA.1/PPR Management of security attributes

FMT_MSA.1.1/PPR The TSF shall enforce the **Security Channel protocol information flow SFP, ISD-P content access control SFP and ISD-R access control SFP** to restrict the ability to change_default, query, modify and delete the security attributes

- **D.PROFILE_POLICY_RULES**

to

- **S.ISD-R to change_default, via function "ES8.ConfigureISDP"**
- **S.ISD-R to query**
- **S.ISD-P to modify, via function "ES6.UpdateMetadata"**
- **S.ISD-R to delete, via function "ES10c.DeleteProfile".**

FMT_MSA.1/CERT_KEYS Management of security attributes

FMT_MSA.1.1/CERT_KEYS The TSF shall enforce the **Security Channel protocol information flow SFP, ISD-R access control SFP and ECASD access control SFP** to restrict the ability to query and delete the security attributes

- **D.CERT.EUICC.ECDSA**
- **D.PK.CI.ECDSA**
- **D.CERT.EUM.ECDSA**
- **D.MNO_KEYS**

to

- **S.ISD-R for:**
  - **query D.PK.CI.ECDSA**
  - **delete D.MNO_KEYS, via function "ES10c.DeleteProfile"**
- **no actor for other operations.**

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: [assignment: **following list of management functions**].

List of management functions:

- SCP information flow control (linked to roles S.ISD-R, U.SM-DPplus, S.ISD-P, U.MNO-SD, U.MNO-OTA)
- Platform services information flow control (linked to roles S.PPI, S.ISD-P, S.ISD-R, U.MNO-SD)
- ISD-R access control (linked to role S.ISD-R, U.SM-DPplus)
- ISD-P content access control (linked to roles S.ISD-P, U.MNO-SD, U.MNO-OTA)
- ECASD access control (linked to roles S.ECASD)

FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles

- **External users:**
  - **U.SM-DPplus**
  - **U.MNO-SD**
  - **U.MNO-OTA**
- **Subjects:**
  - **S.ISD-R**
  - **S.ISD-P**
  - **S.ECASD**
  - **S.PPI**
  - **S.PPE**
  - **S.TELECOM**.

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

FMT_MSA.1/RAT Management of security attributes

FMT_MSA.1.1/RAT The TSF shall enforce the **Platform services information flow SFP and ISD-R access control SFP** to restrict the ability to query the security attributes

- **D.PLATFORM_RAT**

to

- **S.ISD-R to query**
- **S.PPE to query.**

FMT_MSA.3 Static attribute initialisation

FMT_MSA.3.1 The TSF shall enforce the **Security Channel Protocol information flow control SFP, ISD-P content access control SFP, ISD-R access control SFP and ECASD access control SFP** to provide restrictive default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the **no actor** to specify alternative initial values to override the default values when an object or information is created.

7.1.6 Mobile Network authentication

FCS_COP.1/Mobile_network Cryptographic operation

FCS_COP.1.1/Mobile_network the TSF shall perform **Network authentication** in accordance with a specified cryptographic algorithm **MILENAGE, Tuak, [selection: CAVE]** and cryptographic key sizes **according to the corresponding standard** that meet the following:

- **MILENAGE according to standard [20] with the following restrictions:**
  - **Only use 128-bit AES as the kernel function? do not support other choices**
  - **Allow any value for the constant OP**
  - **Allow any value for the constants C1-C5 and R1-R5, subject to the rules and recommendations in section 5.3 of the standard [20]**
- **Tuak according to [21] with the following restrictions:**
  - **Allow any value of TOP**
  - **Allow multiple iterations of Keccak**
  - **Support 256-bit K as well as 128-bit**
  - **To restrict supported sizes for RES, MAC, CK and IK to those currently supported in 3GPP standards.**
- CAVE according to standard TIA TR-45.AHAG Common Cryptographic Algorithms

FCS_CKM.2/Mobile_network Cryptographic key distribution

FCS_CKM.2.1/Mobile_network the TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: following key distribution **methods**] that meets the following: [assignment: **following standards**].

Item	Method	Standard
Milenage	distribution method from SCP-SGP22 (SCP03t)	[SGP.22]
Tuak	distribution method from SCP-SGP22 (SCP03t)	[SGP.22]
CAVE	distribution method from SCP-SGP22 (SCP03t)	[SGP.22]

FCS_CKM.4/Mobile_network Cryptographic key destruction

FCS_CKM.4.1/Mobile_network The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: **clearKey() method**] that meets the following: [assignment: **[JCAPI3]**].

7.2 Runtime Environment Security Requirements

The following conventions for [PP-GP] and [PP-JCS] are used in the definitions of the SFRs:

- Selections and assignments that have already been made in the [PP-GP] and [PP-JCS] Protection Profiles are **in bold**, and the original text on which the selection or assignment has been made is not reminded.
- Selections and assignments made in this ST are **in bold and underlined**, and the PP original text on which the selection or assignment has been made is indicated in a footnote.
- Iteration operations on SFR components are denoted by showing a slash "/" and the iteration indicator after the SFR component identifier.

The Subjects (prefixed with an "S"), the Objects (prefixed with an "O"), Information (prefixed with an "I") are defined and described in [PP-JCS] section 7.1. Security attributes linked to these subjects, objects and information are also defined in [PP-JCS] section 7.1. Finally, Operations (prefixed with "OP") definition and description are present in [PP-JCS] section 7.1.

7.2.1 CoreLG Security Functional requirements

7.2.1.1 Firewall Policy

Firewall Policy

FDP_ACC.2/FIREWALL	Complete access control

FDP_ACC.2.1/FIREWALL The TSF shall enforce the **FIREWALL access control SFP** on **S.CAP_FILE, S.JCRE, S.JCVM, O.JAVAOBJECT** and all operations among subjects and objects covered by the SFP.

Refinement: the operations involved in the policy are : OP.CREATE, OP.INVK_INTERFACE, OP.INVK_VIRTUAL, OP.JAVA, OP.THROW, OP.TYPE_ACCESS, OP.ARRAY_LENGTH, OP.ARRAY_T_ALOAD, OP.ARRAY_T_ASTORE, OP.ARRAY_AASTORE.

FDP_ACC.2.2/FIREWALL The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

Application note: It should be noticed that accessing array's components of a static array, and more generally fields and methods of static objects, is an access to the corresponding O.JAVAOBJECT.

FDP_ACF.1/FIREWALL	Security attribute based access control

FDP_ACF.1.1/FIREWALL The TSF shall enforce the **FIREWALL access control SFP** to objects based on the following:

Subject / Object	Security attributes
S.CAP_FILE	LC Selection Status
S.JCVM	Active Applets, Currently Active Context
S.JCRE	Selected Applet Context
O.JAVAOBJECT	Sharing, Context, LifeTime

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

- **R.JAVA.1 ([JCRE3], §6.2.8): S.CAP_FILE may freely perform OP.INVK_VIRTUAL, OP.INVK_INTERFACE, OP.THROW or OP.TYPE_ACCESS upon any O.JAVAOBJECT whose Sharing attribute has value "JCRE entry point" or "global array".**

- **R.JAVA.2 ([JCRE3], §6.2.8): S.CAP_FILE may freely perform OP.ARRAY_ACCESS, OP.INSTANCE_FIELD, OP.INVK_VIRTUAL, OP.INVK_INTERFACE or OP.THROW upon any O.JAVAOBJECT whose Sharing attribute has value "Standard" and whose Lifetime attribute has value "PERSISTENT" only if O.JAVAOBJECT's Context attribute has the same value as the active context.**
- **R.JAVA.3 ([JCRE3], §6.2.8.10): S.CAP_FILE may perform OP.TYPE_ACCESS upon an O.JAVAOBJECT with Context attribute different from the currently active context, whose Sharing attribute has value "SIO" only if O.JAVAOBJECT is being cast into (checkcast) or is being verified as being an instance of (instanceof) an interface that extends the Shareable interface.**
- **R.JAVA.4 ([JCRE3], §6.2.8.6): S.CAP_FILE may perform OP.INVK_INTERFACE upon an O.JAVAOBJECT with Context attribute different from the currently active context, whose Sharing attribute has the value "SIO", and whose Context attribute has the value "CAP File AID ", only if the invoked interface method extends the Shareable interface and one of the following conditions applies:**
  - **a) The value of the attribute Selection Status of the CAP file whose AID is "CAP File AID" is "Multiselectable",**
  - **b) The value of the attribute Selection Status of the CAP file whose AID is "CAP File AID " is "Non-multiselectable", and either "CAP File AID" is the value of the currently selected applet or otherwise "CAP File AID" does not occur in the attribute Active Applets.**
- **R.JAVA.5: S.CAP_FILE may perform OP.CREATE upon O.JAVAOBJECT only if the value of the Sharing parameter is "Standard" or "SIO".**
- **R.JAVA.6 ([JCRE3], §6.2.8): S.CAP_FILE may freely perform OP.ARRAY_ACCESS or OP.ARRAY_LENGTH upon any O.JAVAOBJECT whose Sharing attribute has value "global array".**

FDP_ACF.1.3/FIREWALL The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

1) **The subject S.JCRE can freely perform OP.JAVA(") and OP.CREATE, with the exception given in FDP_ACF.1.4/FIREWALL, provided it is the Currently Active Context.**
2) **The only means that the subject S.JCVM shall provide for an application to execute native code is the invocation of a Java Card API method (through OP.INVK_INTERFACE or OP.INVK_VIRTUAL).**

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

1) **Any subject with OP.JAVA upon an O.JAVAOBJECT whose LifeTime attribute has value "CLEAR_ON_DESELECT" if O.JAVAOBJECT's Context attribute is not the same as the Selected Applet Context.**
2) **Any subject attempting to create an object by the means of OP.CREATE and a "CLEAR_ON_DESELECT" LifeTime parameter if the active context is not the same as the Selected Applet Context.**
3) **S.CAP_FILE performing OP.ARRAY_AASTORE of the reference of an O.JAVAOBJECT whose sharing attribute has value "global array" or "Temporary".**
4) **S.CAP_FILE performing OP.PUTFIELD or OP.PUTSTATIC of the reference of an O.JAVAOBJECT whose sharing attribute has value "global array" or "Temporary".**
5) **R.JAVA.7 ([JCRE3], §6.2.8.2): S.CAP_FILE performing OP.ARRAY_T_ASTORE into an array view without ATTR_WRITABLE_VIEW access attribute.**
6) **R.JAVA.8 ([JCRE3], §6.2.8.2):S.CAP_FILE performing OP.ARRAY_T_ALOAD into an array view without ATTR_READABLE_VIEW access attribute.**

Application note, FDP_ACF.1.4/FIREWALL:

The deletion of applets may render some O.JAVAOBJECT inaccessible, and the Java Card RE may be in charge of this aspect. This can be done, for instance, by ensuring that references to objects belonging to a deleted application are considered as a null reference. Such a mechanism is implementation-dependent.

In the case of an array type, fields are components of the array ([JVM], §2.14, §2.7.7), as well as the length; the only methods of an array object are those inherited from the Object class.

The Sharing attribute defines five categories of objects:

- Standard ones, whose both fields and methods are under the firewall policy,
- Shareable interface Objects (SIO), which provide a secure mechanism for inter-applet communication,
- JCRE entry points (Temporary or Permanent), who have freely accessible methods but protected fields,
- Global arrays, having both unprotected fields (including components; refer to JavaCardClass discussion above) and methods.
- Array Views, having fields/elements access controlled by access control attributes, ATTR_READABLE_VIEW and ATTR_WRITABLE_VIEW and methods.

When a new object is created, it is associated with the Currently Active Context. Nevertheless, the object is owned by the applet instance within the Currently Active Context when the object is instantiated ([JCRE3], §6.1.3). An object is owned by an applet instance, by the JCRE or by the library where it has been defined (these latter objects can only be arrays that initialize static fields of CAP files).

([JCRE3], Glossary) Selected Applet Context. The Java Card RE keeps track of the currently selected Java Card applet. Upon receiving a SELECT command with this applet's AID, the Java Card RE makes this applet the Selected Applet Context. The Java Card RE sends all APDU commands to the Selected Applet Context.

While the expression "Selected Applet Context" refers to a specific installed applet, the relevant aspect to the policy is the context (CAP file AID) of the selected applet. In this policy, the "Selected Applet Context" is the AID of the selected CAP file.

([JCRE3], §6.1.2.1) At any point in time, there is only one active context within the Java Card VM (this is called the Currently Active Context).

It should be noticed that the invocation of static methods (or access to a static field) is not considered by this policy, as there are no firewall rules. They have no effect on the active context as well and the "acting CAP File" is not the one to which the static method belongs to in this case.

It should be noticed that the Java Card platform, version 2.2.x and version 3.x.x Classic Edition, introduces the possibility for an applet instance to be selected on multiple logical channels at the same time, or accepting other applets belonging to the same CAP file being selected simultaneously. These applets are referred to as multiselectable applets. Applets that belong to a same CAP file are either all multiselectable or not ([JCVM3], §2.2.5). Therefore, the selection mode can be regarded as an attribute of CAP files. No selection mode is defined for a library CAP file.

An applet instance will be considered an active applet instance if it is currently selected in at least one logical channel. An applet instance is the currently selected applet instance only if it is processing the current command. There can only be one currently selected applet instance at a given time. ([JCRE3], §4).

FDP_IFC.1/JCVM	**Subset information flow control**

FDP_IFC.1.1/JCVM The TSF shall enforce the **JCVM information flow control SFP** on **S.JCVM, S.LOCAL, S.MEMBER, I.DATA and OP.PUT(S1, S2, I)**.

Application note: it should be noticed that references of temporary Java Card RE entry points, which cannot be stored in class variables, instance variables or array components, are transferred from the internal memory of the Java Card RE (TSF data) to some stack through specific APIs (Java Card RE owned exceptions) or Java Card RE invoked methods (such as the process (APDU apdu)); these are causes of OP.PUT(S1,S2,I) operations as well.

FDP_IFF.1/JCVM Simple security attributes

FDP_IFF.1.1/JCVM The TSF shall enforce the **JCVM information flow control SFP** based on the following types of subject and information security attributes:

Subjects	Security attributes
S.JCVM	Currently Active Context

FDP_IFF.1.2/JCVM The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:

- **An operation OP.PUT (S1, S.MEMBER, I.DATA) is allowed if and only if the Currently Active Context is "Java Card RE";**
- **Other OP.PUT operations are allowed regardless of the Currently Active Context's value**.

FDP_IFF.1.3/JCVM The TSF shall enforce the **No additional rules**[1].

FDP_IFF.1.4/JCVM The TSF shall explicitly authorize an information flow based on the following rules: **No additional rules**[2].

FDP_IFF.1.5/JCVM The TSF shall explicitly deny an information flow based on the following rules: **No additional rules**[3].

Application note:

The storage of temporary Java Card RE-owned objects references is runtime-enforced ([JCRE3], §6.2.8.1-3).

It should be noticed that this policy essentially applies to the execution of bytecode. Native methods, the Java Card RE itself and possibly some API methods can be granted specific rights or limitations through the FDP_IFF.1.3/JCVM to FDP_IFF.1.5/JCVM elements. The way the Java Card virtual machine manages the transfer of values on the stack and local variables (returned values, uncaught exceptions) from and to internal registers is implementation-dependent. For instance, a returned reference, depending on the implementation of the stack frame, may transit through an internal register prior to being pushed on the stack of the invoker. The returned bytecode would cause more than one OP.PUT operation under this scheme.

FDP_RIP.1/OBJECTS Subset residual information protection

FDP_RIP.1.1/OBJECTS The TSF shall ensure that any previous information content of a resource is made unavailable upon the **allocation of the resource to** the following objects: **class instances and arrays**.

Application note: the semantics of the Java programming language requires for any object field and array position to be initialized with default values when the resource is allocated [JVM], §2.5.1.

FMT_MSA.1/JCRE Management of security attributes

FMT_MSA.1.1/JCRE The TSF shall enforce the **FIREWALL access control SFP** to restrict the ability to **modify** the security attributes **Selected Applet Context** to **the Java Card RE** .

Application note: the modification of the Selected Applet Context should be performed in accordance with the rules given in [JCRE3], §4 and [JCVM3], §3.4.

[1] [assignment: additional information flow control SFP rules]
[2] [assignment: rules, based on security attributes, that explicitly authorize information flows]
[3] [assignment: rules, based on security attributes, that explicitly deny information flows]

FMT_MSA.1/JCVM	Management of security attributes

FMT_MSA.1.1/JCVM The TSF shall enforce the **FIREWALL access control SFP and the JCVM information flow control SFP** to restrict the ability to **modify** the security attributes **Currently Active Context and Active Applets** to **the Java Card VM (S.JCVM)**.

Application note: the modification of the Currently Active Context should be performed in accordance with the rules given in [JCRE3], §4 and [JCVM3], §3.4.

FMT_MSA.2/FIREWALL_JCVM	Secure security attributes

FMT_MSA.2.1/FIREWALL_JCVM The TSF shall ensure that only secure values are accepted for **all the security attributes of subjects and objects defined in the FIREWALL access control SFP and the JCVM information flow control SFP**.

Application note: the following rules are given as examples only. For instance, the last two rules are motivated by the fact that the Java Card API defines only transient arrays factory methods. Future versions may allow the creation of transient objects belonging to arbitrary classes; such evolution will naturally change the range of "secure values" for this component.

- The Context attribute of an O.JAVAOBJECT must correspond to that of an installed applet or be "Java Card RE".
- An O.JAVAOBJECT whose Sharing attribute is a Java Card RE entry point or a global array necessarily has "Java Card RE" as the value for its Context security attribute.
- Any O.JAVAOBJECT whose Sharing attribute value is not "Standard" has a PERSISTENT-LifeTime attribute's value.
- Any O.JAVAOBJECT whose LifeTime attribute value is not PERSISTENT has an array type as JavaCardClass attribute's value.

FMT_MSA.3/FIREWALL	Static attribute initialization

FMT_MSA.3.1/FIREWALL The TSF shall enforce the **FIREWALL access control SFP** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/FIREWALL **[Editorially Refined]** The TSF shall not allow **any role** to specify alternative initial values to override the default values when an object or information is created.

Application note, FMT_MSA.3.1/FIREWALL:

Objects' security attributes of the access control policy are created and initialized at the creation of the object or the subject. Afterwards, these attributes are no longer mutable (FMT_MSA.1/JCRE). At the creation of an object (OP.CREATE), the newly created object, assuming that the FIREWALL access control SFP permits the operation, gets its Lifetime and Sharing attributes from the parameters of the operation; on the contrary, its Context attribute has a default value, which is its creator's Context attribute and AID respectively ([JCRE3], §6.1.3). There is one default value for the Selected Applet Context that is the default applet identifier's Context and one default value for the Currently Active Context that is "Java Card RE".
The knowledge of which reference corresponds to a temporary entry point object or a global array and which does not is solely available to the Java Card RE (and the Java Card virtual machine).

Application note, FMT_MSA.3.2/FIREWALL:

The intent is that none of the identified roles has privileges with regard to the default values of the security attributes. It should be noticed that creation of objects is an operation controlled by the FIREWALL access control SFP. The operation shall fail anyway if the created object would have had security attributes whose value violates FMT_MSA.2.1/FIREWALL_JCVM.

FMT_MSA.3/JCVM	Static attribute initialization

FMT_MSA.3.1/JCVM The TSF shall enforce the **JCVM information flow control SFP** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/JCVM **[Editorially Refined]** The TSF shall not allow **any role** to specify alternative initial values to override the default values when an object or information is created.

FMT_SMF.1/JC	Specification of Management Functions

FMT_SMF.1.1/JC The TSF shall be capable of performing the following management functions: **modify the Currently Active Context, the Selected Applet Context and the Active Applets**.

FMT_SMR.1/JC	Security roles

FMT_SMR.1.1/JC The TSF shall maintain the roles**:**
- **Java Card RE (JCRE),**
- **Java Card VM (JCVM)**.

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

7.2.1.2 Application Programming Interface

FCS_CKM.1/ECDSA	Cryptographic key generation

FCS_CKM.1.1/ECDSA The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **ECDSA Key Pair Generation**[4] and specified cryptographic key sizes **P ranging from 160 to 521 bits**[5] that meet the following: **see application note**[6].

Application note:
- The keys are generated and diversified in accordance with [JCAPI3] in classes KeyBuilder (buildKey method) and KeyPair (genKeyPair method).
- The TOE implements elliptic curve cryptography over GF(p), supporting the following [JCAPI3] key types:

[JCAPI3] class	Supported parameters
javacard.security.KeyBuilder	TYPE_EC_FP_PRIVATE LENGTH_EC_FP_160 TYPE_EC_FP_PRIVATE LENGTH_EC_FP_192 TYPE_EC_FP_PRIVATE LENGTH_EC_FP_224 TYPE_EC_FP_PRIVATE LENGTH_EC_FP_256 TYPE_EC_FP_PRIVATE LENGTH_EC_FP_384 TYPE_EC_FP_PRIVATE LENGTH_EC_FP_521 TYPE_EC_FP_PRIVATE_TRANSIENT_RESET TYPE_EC_FP_PRIVATE_TRANSIENT_DESELECT
javacard.security.KeyPair	ALG_EC_FP LENGTH_EC_FP_160 ALG_EC_FP LENGTH_EC_FP_192 ALG_EC_FP LENGTH_EC_FP_224 ALG_EC_FP LENGTH_EC_FP_256 ALG_EC_FP LENGTH_EC_FP_384

[4] [assignment: cryptographic key generation algorithm]
[5] [assignment: cryptographic key sizes]
[6] [assignment: list of standards]

	ALG_EC_FP LENGTH_EC_FP_521

FCS_CKM.1/GP-SCP Cryptographic key generation

FCS_CKM.1.1/GP-SCP The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **[assignment: cryptographic algorithm]** and specified cryptographic key sizes **[assignment: cryptographic key size]** that meet the following: **[assignment: cryptographic standard]**.

SCP protocol	Cryptographic algorithm	Cryptographic key size	Cryptographic standard
SCP02	TDES 2-keys	112 bits	[GPCS] section E.4.1
SCP03	AES	128, 192, 256 bits	[Amd D] section 6.2.1
SCP81	TDES 3-keys	168 bits	[Amd B] section 3.3.2
SCP81	AES	128 bits	[Amd B] section 3.3.2

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method **see application note**[7] that meets the following: **[JCAPI3] standard**[8].
Application note: the keys are reset as specified in [JCAPI3] Key class, with the method clearKey(). Any access to a cleared key for ciphering or signing shall throw an exception.

FCS_COP.1/TDES_MAC Cryptographic operation

FCS_COP.1.1/TDES_MAC The TSF shall perform **MAC computation of applet instance's data**[9] in accordance with a specified cryptographic algorithm **MAC algorithms mentioned in the application note below**[10] and cryptographic key sizes **112 bits for TDES 2 Keys, 168 bits for TDES 3 Keys**[11] that meet the following: **FIPS PUB 46-3, FIPS PUB 81, ISO/IEC 9797-1, PKCS#5**[12].

Application note: the following TDES MACs from [JCAPI3] are implemented:

MAC length	MAC algorithm	Field name in [JCAPI3] Signature class
4 bytes	ISO9797-1 MAC algorithm 3	ALG_DES_MAC4_ISO9797_1_M1_ALG3
4 bytes	ISO9797-1 MAC algorithm 3	ALG_DES_MAC4_ISO9797_1_M2_ALG3
4 bytes	3DES in outer CBC mode	ALG_DES_MAC4_ISO9797_M1
4 bytes	3DES in outer CBC mode	ALG_DES_MAC4_ISO9797_M2
4 bytes	3DES in outer CBC mode	SIG_CIPHER_DES_MAC4
4 bytes	3DES in outer CBC mode	ALG_DES_MAC4_PKCS5
4 bytes	3DES in outer CBC mode	ALG_DES_MAC4_NOPAD
8 bytes	ISO9797-1 MAC algorithm 3	ALG_DES_MAC8_ISO9797_1_M1_ALG3
8 bytes	ISO9797-1 MAC algorithm 3	ALG_DES_MAC8_ISO9797_1_M2_ALG3

[7] [assignment: cryptographic key destruction method]
[8] [assignment: list of standards]
[9] [assignment: list of cryptographic operations]
[10] [assignment: cryptographic algorithm]
[11] [assignment: cryptographic key sizes]
[12] [assignment: list of standards]

8 bytes	3DES in outer CBC mode	ALG_DES_MAC8_ISO9797_M1
8 bytes	3DES in outer CBC mode	ALG_DES_MAC8_ISO9797_M2
8 bytes	3DES in outer CBC mode	SIG_CIPHER_DES_MAC8
8 bytes	3DES in outer CBC mode	ALG_DES_MAC8_PKCS5
8 bytes	3DES in outer CBC mode	ALG_DES_MAC8_NOPAD

FCS_COP.1/AES_MAC Cryptographic operation

FCS_COP.1.1/AES_MAC The TSF shall perform **MAC computation of applet instance's data**[13] in accordance with a specified cryptographic algorithm **MAC algorithms mentioned in the application note below**[14] and cryptographic key sizes **128, 192 and 256 bits**[15] that meet the following: **FIPS PUB 197, NIST SP800-38A**[16].
Application note: the following AES MACs from [JCAPI3] are implemented:

MAC length	MAC algorithm	Field name in [JCAPI3] Signature class
16 bytes	AES in CBC mode, block size 128 bits	ALG_AES_MAC_128_NOPAD
16 bytes	AES in CBC mode, block size 128 bits	SIG_CIPHER_AES_MAC128
16 bytes	AES in CBC mode, block size 128 bits	SIG_CIPHER_AES_CMAC128
16 bytes	AES in CBC mode, block size 128 bits	ALG_AES_CMAC_128
24 bytes	AES in CBC mode, block size 192 bits	ALG_AES_MAC_192_NOPAD
32 bytes	AES in CBC mode, block size 256 bits	ALG_AES_MAC_256_NOPAD

FCS_COP.1/ECDH Cryptographic operation

FCS_COP.1.1/ECDH The TSF shall perform **Secret Key Agreement**[17] in accordance with a specified cryptographic algorithm **Elliptic Curve Diffie-Hellman (ECDH)**[18] and cryptographic key sizes **P ranging from 160 to 521 bits**[19] that meet the following: **IEEE P1363**[20].

Application note: the secret keys are derived using the KeyAgreement class (generateSecret method) of javacard.security. The following [JCAPI3] fields are supported:

Field name in [JCAPI3] KeyAgreement class
ALG_EC_SVDP_DH
ALG_EC_SVDP_DH_KDF
ALG_EC_SVDP_DH_PLAIN
ALG_EC_SVDP_DH_PLAIN_XY
ALG_EC_SVDP_DHC
ALG_EC_SVDP_DHC_KDF
ALG_EC_SVDP_DHC_PLAIN

FCS_COP.1/CRC Cryptographic operation

[13] [assignment: list of cryptographic operations]
[14] [assignment: cryptographic algorithm]
[15] [assignment: cryptographic key sizes]
[16] [assignment: list of standards]
[17] [assignment: list of cryptographic operations]
[18] [assignment: cryptographic algorithm]
[19] [assignment: cryptographic key sizes]
[20] [assignment: list of standards]

FCS_COP.1.1/CRC The TSF shall perform **Computation of checksum of applet instance's data**[21] in accordance with a specified cryptographic algorithm **CRC16 or CRC32**[22] and cryptographic key sizes **none**[23] that meets the following: **ISO/IEC 3309**[24].

Application note: the related algorithms in [JCAPI3] are ALG_ISO3309_CRC16 and ALG_ISO3309_CRC32 (class Checksum of javacard.security).

FCS_COP.1/ECDSA_SIGN Cryptographic operation

FCS_COP.1.1/ECDSA_SIGN The TSF shall perform **signature generation and signature verification of applet instance's data**[25] in accordance with a specified cryptographic algorithm **ECDSA as mentioned in the application note below**[26] and cryptographic key sizes **P ranging from 160 to 521 bits**[27] that meet the following: **FIPS PUB 186-4**[28].

Application note: the following ECDSA signatures from [JCAPI3] are implemented:

Hash algorithm	Field name in [JCAPI3] Signature class
SHA1	ALG_ECDSA_SHA
SHA224	ALG_ECDSA_SHA_224
SHA256	ALG_ECDSA_SHA_256
SHA384	ALG_ECDSA_SHA_384
SHA512	ALG_ECDSA_SHA_512
-	SIG_CIPHER_ECDSA
-	SIG_CIPHER_ECDSA_PLAIN

FCS_COP.1/ECKA_EG Cryptographic operation

FCS_COP.1.1/ECKA_EG The TSF shall perform **[assignment: key agreement]** in accordance with a specified cryptographic algorithm **[assignment: ECKA-EG algorithm]** and cryptographic key sizes **[assignment: NIST P-256, brainpoolP256r1]** that meet the following: [**assignment: FIPS PUB 186-3 Digital Signature Standard, BSI TR-03111 Version 1.11 RFC 5639**].

[JCAPI3] class	Implemented algorithm
KeyAgreement	ALG_EC_SVDP_DH
	ALG_EC_SVDP_DH_PLAIN

FCS_COP.1/GP-SCP Cryptographic operation

FCS_COP.1.1/GP-SCP The TSF shall perform **[assignment: cryptographic operations]** in accordance with a specified cryptographic algorithm **[assignment: cryptographic algorithms]** and

[21] [assignment: list of cryptographic operations]
[22] [assignment: cryptographic algorithm]
[23] [assignment: cryptographic key sizes]
[24] [assignment: list of standards]
[25] [assignment: list of cryptographic operations]
[26] [assignment: cryptographic algorithm]
[27] [assignment: cryptographic key sizes]
[28] [assignment: list of standards]

cryptographic key sizes **[assignment: cryptographic key sizes]** that meet the following: **[assignment: cryptographic standards]**.

SCP Protocol	Operation	Algorithm	Key Sizes	Standards
SCP03, SCP11	Symmetric Encryption/ Decryption	AES in CBC mode	128, 192, or 256 bits	FIPS 197 NIST 800 38A
SCP03	MAC Generation/ Verification	CMAC AES	128, 192, or 256 bits	NIST 800 38B
SCP03	Key Derivation	CMAC-based KDF using AES	128, 192, or 256 bits	NIST 800 108 NIST 800 38B
SCP03, SCP11	Hash Computing	SHA-256, SHA-384, SHA-512	-	ISO 10118 3 FIPS 180 4
SCP80	Secure communication channel with OTA Server	TDES or AES	TDES: 112 bits AES: 128, 192, or 256 bits	TS 102 225 TS 102 226
SCP81	Secure communication channel with the Remote Administration Server	TLS_PSK_WITH_3DES_EDE_CBC_SHA TLS_PSK_WITH_AES_128_CBC_SHA TLS_PSK_WITH_NULL_SHA TLS_PSK_WITH_AES_128_CBC_SHA256 TLS_PSK_WITH_NULL_SHA256		[Amd B] section 3.3.2
SCP-SGP22	**Secure communication channel with the SM-DP+ for mutual authentication**	**ECKA-EG**	**NIST P-256, brainpoolP256r1**	**SGP.22 FIPS PUB 186-3 Digital Signature Standard, BSI TR-03111 Version 1.11 RFC 5639**
SCP-SGP22 (SCP03t)	**Secure communication channel with the SM-DP+ for profile download**	**AES**	**AES: 128**	**SGP.02**

FCS_COP.1/TDES_CIPHER Cryptographic operation

FCS_COP.1.1/TDES_CIPHER The TSF shall perform **encryption and decryption of applet instance's data**[29] in accordance with a specified cryptographic algorithm **Triple DES 2 Keys or Triple DES 3 Keys with cipher modes mentioned in the application note below**[30] and cryptographic key sizes **112 bits for TDES 2 Keys, 168 bits for TDES 3 Keys**[31] that meet the following: **FIPS PUB 46-3, FIPS PUB 81, ISO/IEC 9797-1, PKCS#5**[32].

Application note: the following TDES ciphers from [JCAPI3] are implemented:

Mode	Field name in [JCAPI3] Cipher class
CBC	ALG_DES_CBC_NOPAD
CBC	ALG_DES_CBC_ISO9797_M1
CBC	ALG_DES_CBC_ISO9797_M2
CBC	ALG_DES_CBC_PKCS5
ECB	ALG_DES_ECB_NOPAD
ECB	ALG_DES_ECB_ISO9797_M1
ECB	ALG_DES_ECB_ISO9797_M2
ECB	ALG_DES_ECB_PKCS5

FCS_COP.1/AES_CIPHER Cryptographic operation

FCS_COP.1.1/AES_CIPHER The TSF shall perform **encryption and decryption of applet instance's data**[33] in accordance with a specified cryptographic algorithm **AES with cipher modes mentioned in the application note below**[34] and cryptographic key sizes **128, 192 and 256 bits**[35] that meet the following: **FIPS PUB 197, NIST SP800-38A, NIST SP800-38D, ISO/IEC 9797-1, PKCS#5**[36].

Application note: the following AES ciphers from [JCAPI3] are implemented:

Mode	Field name in [JCAPI3] Cipher class
CBC	ALG_AES_BLOCK_128_CBC_NOPAD
CBC	ALG_AES_CBC_ISO9797_M1
CBC	ALG_AES_CBC_ISO9797_M2
CBC	ALG_AES_CBC_PKCS5
ECB	ALG_AES_BLOCK_128_ECB_NOPAD
ECB	ALG_AES_ECB_ISO9797_M1
ECB	ALG_AES_ECB_ISO9797_M2
ECB	ALG_AES_ECB_PKCS5
CTR	ALG_AES_CTR
Mode	**Field name in [JCAPI3] AEADCipher class**
Counter with CBC-MAC	ALG_AES_CCM
Counter with CBC-MAC	CIPHER_AES_CCM

[29] [assignment: list of cryptographic operations]
[30] [assignment: cryptographic algorithm]
[31] [assignment: cryptographic key sizes]
[32] [assignment: list of standards]
[33] [assignment: list of cryptographic operations]
[34] [assignment: cryptographic algorithm]
[35] [assignment: cryptographic key sizes]
[36] [assignment: list of standards]

FCS_COP.1/RSA_SIGN	Cryptographic operation

FCS_COP.1.1/RSA_SIGN The TSF shall perform **signature generation and signature verification of applet instance's data**[37] in accordance with a specified cryptographic algorithm **RSA Standard and RSA CRT with hash algorithms and padding schemes mentioned in the application note below**[38] and cryptographic key sizes **1024 to 2048 bits by steps of 32 bits, and 3072 bits**[39] that meet the following: **PKCS#1, PKCS#1-PSS (IEEE 1363-2000), ISO/IEC 9796-2 and RFC2409**[40].

Application note: the following RSA signatures from [JCAPI3] are implemented:

Hash algorithm	Padding scheme	Field name in [JCAPI3] Signature class
SHA224	PKCS#1	ALG_RSA_SHA_224_PKCS1
SHA224	PKCS#1-PSS scheme (IEEE 1363-2000)	ALG_RSA_SHA_224_PKCS1_PSS
SHA256	PKCS#1	ALG_RSA_SHA_256_PKCS1
SHA256	PKCS#1-PSS scheme (IEEE 1363-2000)	ALG_RSA_SHA_256_PKCS1_PSS
SHA384	PKCS#1	ALG_RSA_SHA_384_PKCS1
SHA384	PKCS#1-PSS scheme (IEEE 1363-2000)	ALG_RSA_SHA_384_PKCS1_PSS
SHA512	PKCS#1	ALG_RSA_SHA_512_PKCS1
SHA512	PKCS#1-PSS scheme (IEEE 1363-2000)	ALG_RSA_SHA_512_PKCS1_PSS
SHA1	ISO 9796-2	ALG_RSA_SHA_ISO9796
SHA1	ISO 9796-2	ALG_RSA_SHA_ISO9796_MR
SHA1	PKCS#1	ALG_RSA_SHA_PKCS1
SHA1	PKCS#1-PSS scheme (IEEE 1363-2000)	ALG_RSA_SHA_PKCS1_PSS
SHA1	RFC2409	ALG_RSA_SHA_RFC2409

FCS_COP.1/RSA_CIPHER	Cryptographic operation

FCS_COP.1.1/RSA_CIPHER The TSF shall perform **encryption and decryption of applet instance's data**[41] in accordance with a specified cryptographic algorithm **RSA Standard and RSA CRT as mentioned in the application note below**[42] and cryptographic key sizes **1024 to 2048 bits by steps of 32 bits, and 3072 bits**[43] that meet the following: **PKCS#1, PKCS#1-OAEP scheme (IEEE 1363-2000)**[44].

Application note: the following RSA ciphers from [JCAPI3] are implemented:

[JCAPI3] class	Implemented algorithms
Cipher	ALG_RSA_NOPAD
	ALG_RSA_PKCS1
	ALG_RSA_PKCS1_OAEP

FCS_COP.1/Hash	Cryptographic operation

[37] [assignment: list of cryptographic operations]
[38] [assignment: cryptographic algorithm]
[39] [assignment: cryptographic key sizes]
[40] [assignment: list of standards]
[41] [assignment: list of cryptographic operations]
[42] [assignment: cryptographic algorithm]
[43] [assignment: cryptographic key sizes]
[44] [assignment: list of standards]

FCS_COP.1.1/Hash The TSF shall perform **computation of a hash value for applet instance's data** [45] in accordance with a specified cryptographic algorithm **see application note** [46] and cryptographic key sizes **none**[47] that meets the following: **see application note**[48].

Application note: the following hash algorithms from [JCAPI3] are implemented

Hash algorithm	Field name in [JCAPI3] MessageDigest class	Related Standard
SHA1	ALG_SHA	FIPS 180-4
SHA-224	ALG_SHA_224	FIPS 180-4
SHA-256	ALG_SHA_256	FIPS 180-4
SHA-384	ALG_SHA_384	FIPS 180-4
SHA-512	ALG_SHA_512	FIPS 180-4
SHA3-224	ALG_SHA3_224	FIPS 202
SHA3-256	ALG_SHA3_256	FIPS 202
SHA3-384	ALG_SHA3_384	FIPS 202
SHA3-512	ALG_SHA3_512	FIPS 202

FCS_COP.1/HMAC	Cryptographic operation

FCS_COP.1.1/HMAC The TSF shall perform **computation of a HMAC value for applet instance's data** [49] in accordance with a specified cryptographic algorithm **HMAC with hash algorithms mentioned in the application note below**[50] and cryptographic key sizes **see application note**[51] that meet the following: **rfc2104**[52].

Application note: the following HMAC algorithms from [JCAPI3] are implemented

Hash algorithm used in HMAC computation	Field name in [JCAPI3] Signature class
SHA1	ALG_HMAC_SHA1
SHA256	ALG_HMAC_SHA_256
SHA384	ALG_HMAC_SHA_384
SHA512	ALG_HMAC_SHA_512
-	SIG_CIPHER_HMAC

As mentioned in [JCAPI3] the key can be of any length, but it is strongly recommended that the key is not shorter than the byte length of the hash output used in the HMAC implementation. Keys with length greater than the hash block length are first hashed with the hash algorithm used for the HMAC implementation. As required, the implementation also supports an HMAC key length equal to the length of the supported hash algorithm block size.

FDP_RIP.1/ABORT	Subset residual information protection

[45] [assignment: list of cryptographic operations]
[46] [assignment: cryptographic algorithm]
[47] [assignment: cryptographic key sizes]
[48] [assignment: list of standards]
[49] [assignment: list of cryptographic operations]
[50] [assignment: cryptographic algorithm]
[51] [assignment: cryptographic key sizes]
[52] [assignment: list of standards]

FDP_RIP.1.1/ABORT The TSF shall ensure that any previous information content of a resource is made unavailable upon the **deallocation of the resource from** the following objects: **any reference to an object instance created during an aborted transaction**.

Application note: the events that provoke the de-allocation of a transient object are described in [JCRE3], §5.1.

FDP_RIP.1/APDU Subset residual information protection

FDP_RIP.1.1/APDU The TSF shall ensure that any previous information content of a resource is made unavailable upon the **allocation of the resource to** the following objects: **the APDU buffer**.

Application note: the allocation of a resource to the APDU buffer is typically performed as the result of a call to the process () method of an applet.

FDP_RIP.1/bArray Subset residual information protection

FDP_RIP.1.1/bArray The TSF shall ensure that any previous information content of a resource is made unavailable upon the **deallocation of the resource from** the following objects: **the bArray object**.

Application note: a resource is allocated to the bArray object when a call to an applet's install () method is performed. There is no conflict with FDP_ROL.1 here because of the bounds on the rollback mechanism (FDP_ROL.1.2/FIREWALL): the scope of the rollback does not extend outside the execution of the install () method, and the de-allocation occurs precisely right after the return of it

FDP_RIP.1/GlobalArray Subset residual information protection

FDP_RIP.1.1/GlobalArray (refined) The TSF shall ensure that any previous information content of a resource is made unavailable upon **deallocation of the resource from** *the applet as a result of returning from the process method to* the following objects: **a user Global Array**.

Application note: An array resource is allocated when a call to the API method JCSystem.makeGlobalArray is performed. The Global Array is created as a transient JCRE Entry Point Object ensuring that reference to it cannot be retained by any application. On return from the method, which called JCSystem.makeGlobalArray, the array is no longer available to any applet and is deleted and the memory in use by the array is cleared and reclaimed in the next object deletion cycle.

FDP_RIP.1/KEYS Subset residual information protection

FDP_RIP.1.1/KEYS The TSF shall ensure that any previous information content of a resource is made unavailable upon the **deallocation of the resource from** the following objects: **the cryptographic buffer (D.CRYPTO)**.

Application note: the javacard.security & javacardx.crypto packages do provide secure interfaces to the cryptographic buffer in a transparent way. See javacard.security.KeyBuilder and Key interface of [JCAPI3].

FDP_RIP.1/TRANSIENT Subset residual information protection

FDP_RIP.1.1/TRANSIENT The TSF shall ensure that any previous information content of a resource is made unavailable upon the **deallocation of the resource from** the following objects: **any transient object**.

Application note:

- The events that provoke the de-allocation of any transient object are described in [JCRE3], §5.1.
- The clearing of CLEAR_ON_DESELECT objects is not necessarily performed when the owner of the objects is deselected. In the presence of multiselectable applet instances, CLEAR_ON_DESELECT memory segments may be attached to applets that are active in different logical channels. Multiselectable applet instances within a same CAP file must share the transient memory segment if they are concurrently active ([JCRE3], §4.3).

FDP_ROL.1/FIREWALL	Basic rollback

FDP_ROL.1.1/FIREWALL The TSF shall enforce **the FIREWALL access control SFP and the JCVM information flow control SFP** to permit the rollback of the **operations OP.JAVA and OP.CREATE** on the **object O.JAVAOBJECT**.

FDP_ROL.1.2/FIREWALL The TSF shall permit operations to be rolled back within the **scope of a select (), deselect (), process (), install () or uninstall () call, notwithstanding the restrictions given in [JCRE3], §7.7, within the bounds of the Commit Capacity ([JCRE3], §7.8), and those described in [JCAPI3]**.

Application note: transactions are a service offered by the APIs to applets. It is also used by some APIs to guarantee the atomicity of some operation. This mechanism either is implemented in Java Card platform or relies on the transaction mechanism offered by the underlying platform. Some operations of the API are not conditionally updated, as documented in [JCAPI3] (see for instance, PIN-blocking, PIN-checking, update of Transient objects).

7.2.1.3 Card Security Management

FAU_ARP.1	Security alarms

FAU_ARP.1.1 The TSF shall take **one of the following actions: throw an exception, lock the card session, and reinitialize the Java Card System and its data,** upon detection of a potential security violation.

Refinement: the "potential security violation" stands for one of the following events:

- CAP file inconsistency,
- typing error in the operands of a bytecode,
- applet life cycle inconsistency,
- card tearing (unexpected removal of the Card out of the CAD) and power failure,
- abort of a transaction in an unexpected context, (see abortTransaction(), [JCAPI3] and ([JCRE3], §7.6.2)
- violation of the Firewall or JCVM SFPs,
- unavailability of resources,
- array overflow,
- **GlobalPlatform card state inconsistency**[53]

Application note: in FAU_ARP.1.1, the [assignment: list of other actions] is set to ‘none’, meaning that no other actions are defined in this SFR component.

FDP_SDI.2/DATA	Stored data integrity monitoring and action

FDP_SDI.2.1/DATA The TSF shall monitor user data stored in containers controlled by the TSF for **integrity errors**[54] on all objects, based on the following attributes: **integrity check data**[55].

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall **mute the card and decrease the global fault detection counter. Once the global fault detection counter reaches 0, the card is put in degraded mode.**[56]

[53] [assignment: list of other runtime errors]
[54] [assignment: integrity errors]
[55] [assignment: user data attributes]
[56] [assignment: action to be taken]

Application note: the following data persistently stored by TOE have an integrity check data security attribute:

- Key (i.e. objects instance of classes implemented the interface Key)
- PIN (objects instance of class OwnerPin)
- Package
- GlobalPlatform card state (OP_READY, SECURED, CARD_LOCKED, TERMINATE)

FPR_UNO.1 Unobservability

FPR_UNO.1.1 The TSF shall ensure that **any user**[57] are unable to observe the operation **read, write, cryptographic operations**[58] on **PIN, Key**[59] by **any other user or subject**[60].

FPT_FLS.1/JCS Failure with preservation of secure state

FPT_FLS.1.1/JCS The TSF shall preserve a secure state when the following types of failures occur: **those associated to the potential security violations described in FAU_ARP.1**.

Application note: the Java Card RE Context is the Current context when the Java Card VM begins running after a card reset ([JCRE3], §6.2.3) or after a proximity card (PICC) activation sequence ([JCRE3]). Behavior of the TOE on power loss and reset is described in [JCRE3], §3.6 and §7.1. Behavior of the TOE on RF signal loss is described in [JCRE3], §3.6.1.

FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_TDC.1.1 The TSF shall provide the capability to consistently interpret **the CAP files, the bytecode and its data arguments** when shared between the TSF and another trusted IT product.

FPT_TDC.1.2 The TSF shall use

- **the rules defined in [JCVM3] specification,**
- **the API tokens defined in the export files of reference implementation,**
- **none**[61]

When interpreting the TSF data from another trusted IT product.

Application note: concerning the interpretation of data between the TOE and the underlying Java Card platform, it is assumed that the TOE is developed consistently with the SCP functions, including memory management, I/O functions and cryptographic functions.

7.2.1.4 AID Management

FIA_ATD.1/AID User attribute definition

FIA_ATD.1.1/AID The TSF shall maintain the following list of security attributes belonging to individual users:

- **CAP File AID,**
- **Package AID,**
- **Applet's version number,**
- **Registered applet AID,**
- **Applet Selection Status**.

[57] [assignment: list of users and/or subjects]
[58] [assignment: list of operations]
[59] [assignment: list of objects]
[60] [assignment: list of protected users and/or subjects]
[61] [assignment: list of interpretation rules to be applied by the TSF]

Refinement: "Individual users" stand for applets.

FIA_UID.2/AID	User identification before any action

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

Application note:

- By users here, it must be understood the ones associated to the CAP files (or applets) that act as subjects of policies. In the Java Card System, every action is always performed by an identified user interpreted here as the currently selected applet or the CAP file that is the subject's owner. Means of identification are provided during the loading procedure of the CAP file and the registration of applet instances.
- The role Java Card RE defined in FMT_SMR.1 is attached to an IT security function rather than to a "user" of the CC terminology. The Java Card RE does not "identify" itself to the TOE, but it is part of it.

FIA_USB.1/AID	User-subject binding

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

FIA_USB.1.2/AID The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: **CAP file AID are defined with associated value during loading and with context identifier**[62].

FIA_USB.1.3/AID The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: **None**[63].

Application note: the user is the applet and the subject is the S.CAP_FILE. The subject security attribute "Context" shall hold the user security attribute "CAP file AID".

FMT_MTD.1/JCRE	Management of TSF data

FMT_MTD.1.1/JCRE The TSF shall restrict the ability to **modify** the **list of registered applets' AIDs** to **the JCRE**.

Application note:

- The installer and the Java Card RE manage other TSF data such as the applet life cycle or CAP files, but this management is implementation specific. Objects in the Java programming language may also try to query AIDs of installed applets through the lookupAID(...) API method.
- The installer, applet deletion manager or even the card manager may be granted the right to modify the list of registered applets' AIDs in specific implementations (possibly needed for installation and deletion; see #.DELETION and #.INSTALL).

FMT_MTD.3/JCRE	Secure TSF data

FMT_MTD.3.1/JCRE The TSF shall ensure that only secure values are accepted for **the registered applets' AIDs**.

[62] [assignment: rules for the initial association of attributes]
[63] [assignment: rules for the changing of attributes]

7.2.2 INSTG Security Functional requirements

Refer the following SFRs from [PP-GP]:

- **FDP_ITC.2/GP-ELF** replaces FDP_ITC.2/Installer of [PP-JCS]
- **FMT_SMR.1/GP** replaces FMT_SMR.1/Installer of [PP-JCS]
- **FPT_FLS.1/GP** replaces FPT_FLS.1/Installer of [PP-JCS]
- **FPT_RCV.3/GP** replaces FPT_RCV.3/Installer of [PP-JCS]

7.2.3 ADELG Security Functional Requirements

This group consists of the SFRs related to the deletion of applets and/or packages, enforcing the applet deletion manager (ADEL) policy on security aspects outside the runtime. Deletion is a critical operation and therefore requires specific treatment. This policy is better thought as a frame to be fi006Cled by ST implementers.

FDP_ACC.2/ADEL	Complete access control

FDP_ACC.2.1/ADEL The TSF shall enforce the **ADEL access control SFP** on **S.ADEL, S.JCRE, S.JCVM, O.JAVAOBJECT, O.APPLET and O.CODE_CAP_FILE** and all operations among subjects and objects covered by the SFP.

Refinement: the operations involved in the policy are: OP.DELETE_APPLET, OP.DELETE_CAP_FILE, and OP.DELETE_CAP_FILE_APPLET.

FDP_ACC.2.2/ADEL The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

FDP_ACF.1/ADEL	Security attribute based access control

FDP_ACF.1.1/ADEL The TSF shall enforce the **ADEL access control SFP** to objects based on the following:

Subject / Object	Attributes
S.JCVM	Active Applets
S.JCRE	Selected Applet Context, Registered Applets, Resident CAP files
O.CODE_CAP_FILE	CAP file AID, AIDs of packages within a CAP file, Dependent package AID, Static References
O.APPLET	Applet Selection Status
O.JAVAOBJECT	Owner, Remote

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

- **In the context of this policy, an object O is reachable if and only one of the following conditions hold:**
  1) **the owner of O is a registered applet instance A (O is reachable from A),**
  2) **a static field of a resident package P contains a reference to O (O is reachable from P),**
  3) **there exists a valid remote reference to O (O is remote reachable),**
  4) **there exists an object O' that is reachable according to either (1) or (2) or (3) above and O' contains a reference to O (the reachability status of O is that of O').**
- **The following access control rules determine when an operation among controlled subjects and objects is allowed by the policy:**
  - **R.JAVA.14 ([JCRE3], §11.3.4.2, Applet Instance Deletion): S.ADEL may perform OP.DELETE_APPLET upon an O.APPLET only if,**
    1) **S.ADEL is currently selected,**
    2) **there is no instance in the context of O.APPLET that is active in any logical channel and**
    3) **there is no O.JAVAOBJECT owned by O.APPLET such that either O.JAVAOBJECT is reachable from an applet instance distinct from O.APPLET, or**

O.JAVAOBJECT is reachable from a package P, or ([JCRE3], §8.5) O.JAVAOBJECT is remote reachable.

- **R.JAVA.15 ([JCRE3], §11.3.4.2.1, Multiple Applet Instance Deletion): S.ADEL may perform OP.DELETE_APPLET upon several O.APPLET only if,**
  1) **S.ADEL is currently selected,**
  2) **there is no instance of any of the O.APPLET being deleted that is active in any logical channel and**
  3) **there is no O.JAVAOBJECT owned by any of the O.APPLET being deleted such that either O.JAVAOBJECT is reachable from an applet instance distinct from any of those O.APPLET, or O.JAVAOBJECT is reachable from a CAP file P, or ([JCRE3], §8.5) O.JAVAOBJECT is remote reachable.**
- **R.JAVA.16 ([JCRE3], §11.3.4.3, Applet/Library CAP file Deletion): S.ADEL may perform OP.DELETE_CAP_FILE upon an O.CODE_CAP_FILE only if,**
  1) **S.ADEL is currently selected,**
  2) **no reachable O.JAVAOBJECT, from a CAP file distinct from O.CODE_CAP_FILE that is an instance of a class that belongs to O.CODE_CAP_FILE, exists on the card and**
  3) **there is no resident package on the card that depends on O.CODE_CAP_FILE.**
- **R.JAVA.17 ([JCRE3], §11.3.4.4, Applet CAP file and Contained Instances Deletion): S.ADEL may perform OP.DELETE_CAP_FILE_APPLET upon an O.CODE_CAP_FILE only if,**
  1) **S.ADEL is currently selected,**
  2) **no reachable O.JAVAOBJECT, from a CAP file distinct from O.CODE_CAP_FILE, which is an instance of a class that belongs to O.CODE_CAP_FILE exists on the card,**
  3) **there is no CAP file loaded on the card that depends on O.CODE_CAP_FILE, and**
  4) **for every O.APPLET of those being deleted it holds that: (i) there is no instance in the context of O.APPLET that is active in any logical channel and (ii) there is no O.JAVAOBJECT owned by O.APPLET such that either O.JAVAOBJECT is reachable from an applet instance not being deleted, or O.JAVAOBJECT is reachable from a CAP file not being deleted, or ([JCRE3], §8.5) O.JAVAOBJECT is remote reachable.**

FDP_ACF.1.3/ADEL The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: **none**.

FDP_ACF.1.4/ADEL The TSF shall explicitly deny access of subjects to objects based on the following additional rules: **any subject but S.ADEL to O.CODE_PKG or O.APPLET for the purpose of deleting them from the card**.

Application note, FDP_ACF.1.2/ADEL:

- This policy introduces the notion of reachability, which provides a general means to describe objects that are referenced from a certain applet instance or CAP file.
- S.ADEL calls the "uninstall" method of the applet instance to be deleted, if implemented by the applet, to inform it of the deletion request. The order in which these calls and the dependencies checks are performed are out of the scope of this security target.

FDP_RIP.1/ADEL Subset residual information protection

FDP_RIP.1.1/ADEL The TSF shall ensure that any previous information content of a resource is made unavailable upon the **deallocation of the resource from** the following objects: **applet instances and/or CAP files when one of the deletion operations in FDP_ACC.2.1/ADEL is performed on note**
them.

Application note: deleted freed resources (both code and data) may be reused, depending on the way they were deleted (logically or physically). Requirements on de-allocation during applet/CAP file deletion are described in [JCRE3], §11.3.4.1, §11.3.4.2 and §11.3.4.3.

FMT_MSA.1/ADEL Management of security attributes

FMT_MSA.1.1/ADEL The TSF shall enforce the **ADEL access control SFP** to restrict the ability to **modify** the security attributes **Registered Applets and Resident CAP files** to **the Java Card RE**.

Application note: patch deletion is an extension of applet/package deletion defined in GlobalPlatform as a patch is managed as a JavaCard Package and registered with specific attributes.

FMT_MSA.3/ADEL	Static attribute initialization

FMT_MSA.3.1/ADEL The TSF shall enforce the **ADEL access control SFP** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/ADEL The TSF shall allow the **following role(s): none,** to specify alternative initial values to override the default values when an object or information is created.

Application note: patch deletion is an extension of applet/package deletion defined in GlobalPlatform as a patch is managed as a JavaCard Package and registered with specific attributes.

FMT_SMF.1/ADEL	Specification of Management Functions

FMT_SMF.1.1/ADEL The TSF shall be capable of performing the following management functions: **modify the list of registered applets' AIDs and the Resident CAP files**.

FMT_SMR.1/ADEL	Security roles

FMT_SMR.1.1/ADEL The TSF shall maintain the roles**: applet deletion manager**.

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

FPT_FLS.1/ADEL	Failure with preservation of secure state

FPT_FLS.1.1/ADEL The TSF shall preserve a secure state when the following types of failures occur: **the applet deletion manager fails to delete a CAP file/applet as described in [JCRE3], §11.3.4**.

Application note:

- The TOE may provide additional feedback information to the card manager in case of a potential security violation (see FAU_ARP.1).
- The CAP file/applet instance deletion must be atomic. The "secure state" referred to in the requirement must comply with Java Card specification ([JCRE3], §11.3.4.)

Application note: patch deletion is an extension of applet/package deletion defined in GP as a patch is managed as a JavaCard Package and registered with specific attributes.

7.2.4 RMIG Security Functional Requirements

The product does not support RMI features.

7.2.5 ODELG Security Functional Requirements

The following requirements concern the object deletion mechanism. This mechanism is triggered by the applet that owns the deleted objects by invoking a specific API method.

FDP_RIP.1/ODEL	Subset residual information protection

FDP_RIP.1.1/ODEL The TSF shall ensure that any previous information content of a resource is made unavailable upon the **deallocation of the resource from** the following objects: **the objects owned by the context of an applet instance which triggered the execution of the method javacard.framework.JCSystem.requestObjectDeletion()**.

Application note:

- Freed data resources resulting from the invocation of the method javacard.framework.JCSystem.requestObjectDeletion() may be reused. Requirements on de-allocation after the invocation of the method are described in [JCAPI3].
- There is no conflict with FDP_ROL.1 here because of the bounds on the rollback mechanism: the execution of requestObjectDeletion() is not in the scope of the rollback because it must be performed in between APDU command processing, and therefore no transaction can be in progress.

FPT_FLS.1/ODEL	Failure with preservation of secure state

FPT_FLS.1.1/ODEL The TSF shall preserve a secure state when the following types of failures occur: **the object deletion functions fail to delete all the unreferenced objects owned by the applet that requested the execution of the method**.

Application note: the TOE may provide additional feedback information to the card manager in case of potential security violation (see FAU_ARP.1).

7.2.6 CARG Security Functional Requirements

Refer the following SFRs from [PP-GP]:

- **FCO_NRO.2/GP** replaces FCO_NRO.2.1/CM of [PP-JCS]
- **FDP_IFC.2/GP-ELF** replaces FDP_IFC.2/CM of [PP-JCS]
- **FDP_IFF.1/GP-ELF** replaces FDP_IFF.1/CM of [PP-JCS]
- **FDP_UIT.1/GP** replaces FDP_UIT.1/CM of [PP-JCS]
- **FIA_UID.1/GP** replaces FIA_UID.1/CM of [PP-JCS]
- **FMT_MSA.1/GP** replaces FMT_MSA.1/CM of [PP-JCS]
- **FMT_MSA.3/GP** replaces FMT_MSA.3/CM of [PP-JCS]
- **FMT_SMF.1/GP** replaces FMT_SMF.1/CM of [PP-JCS]
- **FTP_ITC.1.3/GP** replaces FTP_ITC.1/CM of [PP-JCS]

7.2.7 Global Platform Security Functional requirements

FPT_FLS.1/GP	Failure with preservation of secure state

FPT_FLS.1.1/GP The TSF shall preserve a secure state when the following types of failures occur:

- **S.OPEN fails to load/install an Executable Load File / Application instance.**
- **S.SD fails to load SD/Application data and keys.**
- **S.OPEN fails to verify/change the Card Life Cycle, Application and SD Life Cycle states.**
- **S.OPEN fails to verify the privileges belonging to an SD or an Application.**
- **S.SD fails to verify the security level applied to protect APDU commands.**
- **None**[64]

Application Note:

- This SFR extends FPT_FLS.1/Installer of [PP-JCS] to include the failures that may occur during the loading of SD/Application keys and data.
- Refer to [JCRE3] section 11.1.5 and [GPCS] sections 11.5, 11.6, 11.8, and 11.11 for additional details.

FDP_ROL.1/GP	Basic rollback

FDP_ROL.1.1/GP The TSF shall enforce **ELF Loading information flow control SFP and Data & Key Loading information flow control SFP** to permit the rollback of the **installation, loading, or removal operation** on the **executable files, application instances, SD/Application data and keys**.

FDP_ROL.1.2/GP The TSF shall permit operations to be rolled back within the **boundary limit**:

- **Until the Executable File or application instance has been added to or removed from the applet's registry.**
- **Until SD/Application data or keys have been added to or removed from SD or Application.**

FCO_NRO.2/GP	Enforced proof of origin

FCO_NRO.2.1/GP The TSF shall enforce the generation of evidence of origin for transmitted **Executable Load Files, SD/Application data and keys** [65] at all times.

Refinement: the TSF shall be able to generate an evidence of origin at all times for 'Executable Load Files, SD/Application data and keys' received from the off-card entity (originator of transmitted data) that communicates with the card.

FCO_NRO.2.2/GP The TSF shall be able to relate the **identity**[66] of the originator of the information, and the **Executable Load Files, SD/Application data and keys**[67] of the information to which the evidence applies.

Refinement: the TSF shall be able to load 'Executable Load Files, SD/Application data and keys' to the card with associated security attributes (the identity of the originator, the destination) such that the evidence of origin can be verified.

FCO_NRO.2.3/GP The TSF shall provide a capability to verify the evidence of origin of information to **the off card entity (recipient of the evidence of origin) who requested that verification** given **at the time the ELF, SD/Application data and keys are received**[68].

---

[64] [assignment: list of additional types of failures]
[65] [assignment: list of information types]
[66] [assignment: list of attributes]
[67] [assignment: list of information fields]
[68] [assignment: limitations on the evidence of origin]

Application Note:
- This SFR extends FCO_NRO.2/CM of [PP-JCS] to cover the SD/Application data and keys transmitted and loaded to the card via STORE DATA and PUT KEY commands.

FMT_SMR.1/GP Security roles

FMT_SMR.1.1/GP The TSF shall maintain the roles:
- **On-card: S.OPEN, S.SD (e.g. ISD, APSD, CASD), Application**
- **Off-card: Issuer, Users (e.g. VA, AP, CA) owning SDs**

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

Application Note: this SFR refines and replaces FMT_SMR.1/Installer and FMT_SMR.1/CM of [PP-JCS], applied to roles involved in card content management operations.

FMT_SMF.1/GP Specification of Management Functions

FMT_SMF.1.1/GP The TSF shall be capable of performing the following management functions **specified in [GPCS]:**
- **Card and Application Security Management as defined in [GPCS]: Life Cycle, Privileges, Application/SD Locking and Unlocking, Card Locking and Unlocking, Card Termination, Application Status interrogation, Card Status Interrogation, command dispatch, Operational Velocity Checking.**
- **Management functions (Secure Channel Initiation/Operation/Termination) related to SCPs as defined in [GPCS].**

Application Note:
- This SFR refines and replaces FMT_SMF.1/CM of [PP-JCS].
- Management functions related to SCPs are defined in [GPCS] Chapter 10.

FDP_ITC.2/GP-ELF Import of user data with security attributes

FDP_ITC.2.1/GP-ELF The TSF shall enforce the **ELF Loading information flow control SFP** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/GP-ELF The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/GP-ELF The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

FDP_ITC.2.4/GP-ELF The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5/GP-ELF The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:
- **Referring to Java Card rules defined in [JCVM3] and [JCRE3]: ELF loading is allowed only if, for each dependent ELF, its AID attribute is equal to a resident ELF AID attribute, and the major (minor) Version attribute associated with the dependent ELF is less than or equal to the major (minor) Version attribute associated with the resident ELF.**
- **None**[69]

[69] [assignment: additional importation control rules]

Application Note:
- This SFR corresponds to FDP_ITC.2/Installer of [PP-JCS].
- Java Card rules are defined in [JCVM3] sections 4.4 and 4.5 and [JCRE3] section 11.
- The TSF shall use the INSTALL data format and the LOAD data format when interpreting the user data from outside the TOE.

FDP_IFC.2/GP-KL	Complete information flow control

FDP_IFC.2.1/GP-KL The TSF shall enforce the **Data & Key Loading information flow control SFP** on
- **Subjects: S.SD, S.CAD, S.OPEN, Application**
- **Information: GlobalPlatform APDU commands STORE DATA and PUT KEY, GlobalPlatform APIs for loading and storing data and keys**

and all operations that cause that information to flow to and from subjects covered by the SFP.

FDP_IFC.2.2/GP-KL The TSF shall ensure that all operations that cause any information in the TOE to flow to and from any subject in the TOE are covered by an information flow control SFP.

Application Note:
- GlobalPlatform's card content management APDU commands and API methods are described in [GPCS] Chapter 11 and Appendix A.1, respectively.
- The subject S.SD can be the ISD, an APSD, or the CASD.

FPT_RCV.3/GP	Automated recovery without undue loss

FPT_RCV.3.1/GP When automated recovery from **none, see application note below**[70] is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT_RCV.3.2/GP For **detection of a potential loss of integrity during the transmission of an Executable Load File to the card, abortion of the installation process of an Executable Load File, or any fatal error occurred during the linking of an Executable Load File to the Executable Files already installed on the card**[71] the TSF shall ensure the return of the TOE to a secure state using automated procedures.

FPT_RCV.3.3/GP The functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding **the loss of the Executable Load File being loaded or installed**[72] for loss of TSF data or objects under the control of the TSF.

FPT_RCV.3.4/GP The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.

Application Note:
- This SFR refines and replaces FPT_RCV.3/Installer of [PP-JCS], applied to card content management operations
- There is no maintenance mode implemented within the TOE. Recovery is always enforced automatically as stated in FPT_RCV.3.2/GP

[70] [assignment: list of failures/service discontinuities during card content management operations]
[71] [assignment: list of failures/service discontinuities during card content management operations]
[72] [assignment: quantification]

FDP_IFC.2/GP-ELF Complete information flow control

FDP_IFC.2.1/GP-ELF The TSF shall enforce the **ELF Loading information flow control SFP** on
- **Subjects: S.SD, S.CAD, S.OPEN**
- **Information: APDU commands INSTALL and LOAD, GlobalPlatform APIs for loading and installing ELF**

Moreover, all operations that cause that information to flow to and from subjects covered by the SFP.

FDP_IFC.2.2/GP-ELF The TSF shall ensure that all operations that cause any information in the TOE to flow to and from any subject in the TOE are covered by an information flow control SFP.

Application Note:
- This SFR replaces FDP_IFC.2/CM of [PP-JCS].
- The subject S.SD can be the ISD, an APSD, or the CASD.
- GlobalPlatform's card content management APDU commands and API methods are described in [GPCS] Chapter 11 and Appendix A.1, respectively.

FDP_IFF.1/GP-ELF Complete information flow control

FDP_IFF.1.1/GP-ELF The TSF shall enforce the **ELF Loading information flow control SFP** based on the following types of subject and information security attributes:
- **Subjects: S.SD, S.OPEN**
- **Information: APDU commands INSTALL and LOAD, GlobalPlatform APIs for loading and installing ELF**
- **Security attributes: Card Life Cycle state, ELF signature verification status, ELF AID, SD privileges, Secure Channel Security Level**[73].

FDP_IFF.1.2/GP-ELF The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:
- **S.SD implements one or more Secure Channel Protocols, namely SCP02, SCP03, SCP11, SCP80, SCP81**[74]**, each with a complete Secure Channel Key Set.**
- **S.SD has all of the cryptographic keys required by its privileges (e.g. CLFDB, DAP, DM).**
- **On receipt of INSTALL or LOAD commands, S.OPEN checks that the card Life Cycle State is not CARD_LOCKED or TERMINATED.**
- **S.OPEN accepts an ELF only if its integrity and authenticity has been verified.**
- **S.OPEN accepts an ELF only if its AID is not already registered by the TSF**[75]

FDP_IFF.1.3/GP-ELF The TSF shall enforce the **none**[76].

FDP_IFF.1.4/GP-ELF The TSF shall explicitly authorize an information flow based on the following rules: **none**[77].

FDP_IFF.1.5/GP-ELF The TSF shall explicitly deny an information flow based on the following rules:
- **S.OPEN fails to verify the integrity and request verification of the authenticity for ELFs**
- **S.OPEN fails to verify the Card Life Cycle state**
- **S.OPEN fails to verify the SD privileges.**
- **S.SD fails to verify the security level applied to protect INSTALL or LOAD commands.**
- **S.SD fails to set the security level (integrity and/or confidentiality), to apply to the next incoming command and/or next outgoing response.**
- **S.SD fails to unwrap INSTALL or LOAD commands.**

[73] [assignment: list of subjects and information controlled under the indicated SFP, and for each, the security attributes]
[74] [selection: SCP02, SCP03, SCP10, SCP11, SCP21, SCP22, SCP80, SCP81]
[75] [assignment: for each operation, the security attribute-based relationship that must hold between subject and information security attributes]
[76] [assignment: additional information flow control SFP rules]
[77] [assignment: rules, based on security attributes, that explicitly authorize information flows]

- **The ELF AID is already registered within the card**[78]

Application Note:
- This SFR refines and replaces FDP_IFF.1/CM of [PP-JCS].
- APDUs belonging to the policy ELF Loading information flow control SFP are described in the following references:
  - For INSTALL, see [GPCS] section 11.5.
  - For LOAD, see [GPCS] section 11.6.
- The INSTALL and LOAD commands must only be issued within a Secure Channel Session; the levels of security for these commands depend on the security level defined in the EXTERNAL AUTHENTICATE command.
- The Minimum Security Level of INSTALL and LOAD is 'AUTHENTICATED' as defined in [GPCS] section 10.6.
- For more details about the rules to be applied to each role of INSTALL command, refer to [GPCS] sections 9.3 and 3.4.

FIA_UID.1/GP Timing of identification

FIA_UID.1.1/GP The TSF shall allow **SD selection, Application selection, initializing a Secure Channel with the card, requesting data that identifies the card or off-card entities**[79] on behalf of the user to be performed before the user is identified.

FIA_UID.1.2/GP The TSF shall require each user to be successfully identified before allowing any other TSF mediated actions on behalf of that user.

Application Note:
- This SFR refines and replaces FIA_UID.1/CM of [PP-JCS].

FIA_AFL.1/GP Authentication failure handling

FIA_AFL.1.1/GP The TSF shall detect when **1**[80] unsuccessful authentication attempt occur related to **the authentication of the origin of a card management operation command**.

FIA_AFL.1.2/GP When the defined number of unsuccessful authentication attempts has been **met or surpassed**, the TSF shall **close the Secure Channel**.

FIA_UAU.1/GP Timing of authentication

FIA_UAU.1.1/GP The TSF shall allow **the TSF mediated actions listed in FIA_UID.1/GP** on behalf of the user to be performed before the user is authenticated.

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

[78] [assignment: rules, based on security attributes, that explicitly deny information flows]
[79] [assignment: list of TSF-mediated actions]
[80] [selection: [assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]]

FIA_UAU.4/GP Single-use authentication mechanisms

FIA_UAU.4.1/GP The TSF shall prevent reuse of authentication data related to **the authentication mechanism used to open a secure communication channel with the card**.

FDP_UIT.1/GP Basic data exchange integrity

FDP_UIT.1.1/GP The TSF shall enforce the **ELF Loading information flow control SFP and Data & Key Loading information flow control SFP** to **receive**[81] user data in a manner protected from **modification, deletion, insertion, replay** errors.

FDP_UIT.1.2/GP The TSF shall be able to determine on receipt of user data, whether **modification, deletion, insertion, replay** has occurred.

Application Note:
- This SFR extends FDP_UIT.1/CM of [PP-JCS] to cover the integrity protection of SD/Application data and keys.
- This SFR applies where APDU command and response integrity protection is required (e.g. INSTALL, LOAD, STORE DATA and PUT KEY commands).

FDP_UCT.1/GP Basic data exchange confidentiality

FDP_UCT.1.1/GP The TSF shall enforce the **ELF Loading information flow control SFP and Data & Key Loading information flow control SFP** to **receive**[82] user data in a manner protected from unauthorized disclosure.

Application Note: this SFR applies where APDU command and response confidentiality protection is required. For example, the sensitive data (e.g. secret keys) shall always be transmitted as confidential data.

FTP_ITC.1/GP Inter-TSF trusted channel

FTP_ITC.1.1/GP The TSF shall provide a communication channel between itself and another trusted IT product 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/GP The TSF shall permit **another trusted IT product** to initiate communication via the trusted channel.

FTP_ITC.1.3/GP The TSF shall initiate communication via the trusted channel for:
- **APDU commands sent to the card within a Secure Channel Session**
- **When loading/installing a new ELF on the card**
- **When transmitting and loading sensitive data to the card using STORE DATA or PUT KEY commands**
- **When deleting ELFs, Applications, or Keys**
- **None**[83]

Application Note: this SFR corresponds to FTP_ITC.1/CM of [PP-JCS], applied where APDU command and response integrity and/or confidentiality protection through a Secure Channel are required.

[81] [selection: transmit, receive]
[82] [selection: transmit, receive]
[83] [assignment: list of functions for which a trusted channel is required]

FPR_UNO.1/GP	Unobservability

FPR_UNO.1.1/GP The TSF shall ensure that **SDs and Applications** are unable to observe the operation **keys or data import (PUT KEY or STORE DATA), encryption, decryption, signature generation and verification, none**[84] on **keys and data** by **the OPEN or any other SD or Application**.

FPT_TDC.1/GP	Inter-TSF basic TSF data consistency

FPT_TDC.1.1/GP The TSF shall provide the capability to consistently interpret **ELFs, SD/Application data and keys, data used to implement a Secure Channel, None**[85] when shared between the TSF and another trusted IT product.

FPT_TDC.1.2/GP The TSF shall use **the list of interpretation rules to be applied by the TSF when processing the INSTALL, LOAD, PUT KEY, and STORE DATA commands sent to the card, None**[86] when interpreting the TSF data from another trusted IT product.

Application Note: the list of interpretation rules to be applied by the TSF when processing the INSTALL, LOAD, PUT KEY, and STORE DATA commands sent to the card are defined in [GPCS] sections 11.5, 11.6, 11.8, and 11.11.

FDP_ITC.2/GP-KL	Import of user data with security attributes

FDP_ITC.2.1/GP-KL The TSF shall enforce the **Data & Key Loading information flow control SFP** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.2.2/GP-KL The TSF shall use the security attributes associated with the imported user data.

FDP_ITC.2.3/GP-KL The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

FDP_ITC.2.4/GP-KL The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.

FDP_ITC.2.5/GP-KL The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:

- **The algorithms and key sizes of the imported keys shall be supported by the SE**
- **The Key Identifier (Key ID) of the imported keys shall be in an allowed range as specified in section 4 of [CIC]**[87]

Application Note:

- The algorithms and key sizes of the imported keys shall be supported by the Card as specified in [GPCS] Appendices B and C.

PUT KEY and STORE DATA are described in [GPCS] sections 11.8 and 11.11.

FDP_IFF.1/GP-KL	Complete information flow control

FDP_IFF.1.1/GP-KL The TSF shall enforce the **Data & Key Loading information flow control SFP** based on the following types of subject and information security attributes:

- **Subjects: S.SD, S.OPEN**

[84] [assignment: list of operations]
[85] [assignment: list of TSF data types]
[86] [assignment: list of interpretation rules to be applied by the TSF]
[87] [assignment: additional importation control rules]

- **GlobalPlatform APDU commands STORE DATA and PUT KEY, GlobalPlatform APIs for loading and storing data and keys**
- **Security attributes: card Life Cycle State, Application and SD Life Cycle states, Secure Channel Security Level, SD and Application privileges**[88].

FDP_IFF.1.2/GP-KL The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:
- **S.SD implements one or more Secure Channel Protocols, namely SCP02, SCP03, SCP11**[89]**, each equipped with a complete Secure Channel Key Set.**
- **S.SD has all of the cryptographic keys required by its privileges (e.g. CLFDB, DAP, DM).**
- **An Application accepts a message only if it comes from the S.SD it belongs to.**
- **On receipt of a request to forward STORE DATA or PUT KEY commands to an Application, S.OPEN checks that the card Life Cycle State is not CARD_LOCKED or TERMINATED.**
- **On receipt of a request to forward STORE DATA or PUT KEY commands to an Application, the S.OPEN checks that the requesting S.SD has no restrictions for personalization.**
- **S.SD unwraps STORE DATA or PUT KEY according to the Current Security Level of the current Secure Channel Session and prior to the command forwarding to the targeted Application or SD.**
- **S.OPEN verifies that the targeted application implements a personalization interface**[90]

FDP_IFF.1.3/GP-KL The TSF shall enforce the **none**[91].

FDP_IFF.1.4/GP-KL The TSF shall explicitly authorize an information flow based on the following rules: **none**[92].

FDP_IFF.1.5/GP-KL The TSF shall explicitly deny an information flow based on the following rules:
- **S.OPEN fails to verify the Card Life Cycle, Application and SD Life Cycle states.**
- **S.OPEN fails to verify the privileges belonging to an SD or an Application.**
- **S.SD fails to unwrap STORE DATA or PUT KEY.**
- **S.SD fails to verify the security level applied to protect APDU commands.**
- **S.SD fails to set the security level (integrity and/or confidentiality), to apply to the next incoming command and/or next outgoing response.**
- **S.OPEN fails to verify that the targeted application implements a personalization interface**.[93]

Application Note:
- APDUs belonging to the Data & Key Loading information flow control SFP are described in the following references:
  - For PUT KEY, see [GPCS] section 11.8.
  - For STORE DATA, see [GPCS] section 11.11.
- The PUT KEY and STORE DATA commands must only be issued within a Secure Channel Session; the levels of security for these commands depend on the security level defined in the EXTERNAL AUTHENTICATE command.
- The Minimum Security Level of PUT KEY and STORE DATA is 'AUTHENTICATED' as defined in [GPCS] Section 10.6.
- For more details about Key Access Conditions, Data and Key Management, refer to [GPCS] sections 7.5.2 and 7.6.

[88] [assignment: list of subjects and information controlled under the indicated SFP, and for each, the security attributes]
[89] [selection: SCP02, SCP03, SCP10, SCP11, SCP21, SCP22, SCP80, SCP81]
[90] [assignment: for each operation, the security attribute-based relationship that must hold between subject and information security attributes]
[91] [assignment: additional information flow control SFP rules]
[92] [assignment: rules, based on security attributes, that explicitly authorize information flows]
[93] [assignment: rules, based on security attributes, that explicitly deny information flows]

FMT_MSA.1/GP Management of security attributes

FMT_MSA.1.1/GP The TSF shall enforce the **ELF Loading information flow control SFP and Data & Key Loading information flow control SFP** to restrict the ability to **[selection: [assignment: perform the operations listed in table acting on]]** the security attributes **[assignment: mentioned in table]** to **[assignment: the authorized identified roles mentioned in table].**

Operations (APDUs or APIs)	**Security Attributes: Card Life Cycle State**	**Authorised Identified Roles with Privileges**
DELETE Executable Load File	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD
DELETE Executable Load File and related Application(s)	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD
DELETE Application	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD
DELETE Key	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD, SD
INSTALL	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD
INSTALL [for personalisation]	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD, SD
LOAD	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD
PUT KEY	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD, SD
SELECT	OP_READY, INITIALIZED, SECURED	ISD, AM SD, DM SD,
SET STATUS	OP_READY, INITIALIZED, SECURED, or CARD_LOCKED	ISD, AM SD, DM SD, SD
STORE DATA	OP_READY, INITIALIZED, or SECURED	ISD, AM SD, DM SD, SD
GET DATA	OP_READY, INITIALIZED, SECURED, CARD_LOCKED, or TERMINATED	ISD, AM SD, DM SD, SD
GET STATUS	OP_READY, INITIALIZED, SECURED, or CARD_LOCKED	ISD, AM SD, DM SD, SD

Operations: SCP11 Commands	Security Attributes: Card Life Cycle State	Security Attributes: Minimum Security Level	Authorised Identified Roles with Privileges
GET DATA (ECKA Certificate)	OP_READY, INITIALIZED, SECURED, or CARD_LOCKED	None	ISD, AM SD, DM SD, SD
PERFORM SECURITY OPERATION		None	
MUTUAL AUTHENTICATE		AUTHENTICATED or ANY_AUTHENTICATED	
INTERNAL AUTHENTICATE		AUTHENTICATED or ANY_AUTHENTICATED	
STORE DATA (ECKA Certificate)		None	
STORE DATA (Whitelist)		None	
VERIFY PIN		None	

Operations: **SCP80 Command**	**Security Attributes:** **Card Life Cycle State**	**Security Attributes:** **Minimum Security Level**	**Authorised Identified Roles with Privileges**
Remote File Management Commands SELECT, UPDATE BINARY, UPDATE RECORD, SEARCH RECORD, INCREASE, VERIFY PIN, CHANGE PIN, DISABLE PIN, ENABLE PIN, UNBLOCK PIN, DEACTIVATE FILE, ACTIVATE FILE, READ BINARY, READ RECORD, CREATE FILE, DELETE FILE, RESIZE FILE, SET DATA, RETRIEVE DATA	See [TS 102 225] and [TS 102 226]	See [TS 102 225] and [TS 102 226]	See [TS 102 225] and [TS 102 226]
Remote Applet Management Commands DELETE, SET STATUS, INSTALL, LOAD, PUT KEY, GET STATUS, GET DATA, STORE DATA	See [TS 102 225] and [TS 102 226]	See [TS 102 225] and [TS 102 226]	See [TS 102 225] and [TS 102 226]

Operations: SCP81 Command	Security Attributes: Card Life Cycle State	Security Attributes: Minimum Security Level	Authorised Identified Roles with Privileges
PUT KEY	OP_READY, INITIALIZED, SECURED	None	ISD, AM SD, DM SD, SD
STORE DATA	OP_READY, INITIALIZED, SECURED	None	ISD, AM SD, DM SD, SD
GET DATA	OP_READY, INITIALIZED, SECURED, CARD_LOCKED, or TERMINATED	None	ISD, AM SD, DM SD, SD

Legend for tables above:
- ISD: Issuer Security Domain
- AM SD: Security Domain with Authorized Management privilege
- DM SD: Security Domain with Delegated Management privilege
- SD: Other Security Domain

Application Note:
- This SFR refines and replaces FMT_MSA.1/CM of [PP-JCS]. It is extended to cover Data and Key loading Policy.
- The authorized identified roles could be off-card or on-card entities as defined in FMT_SMR.1/GP.

FMT_MSA.3/GP Security attribute initialization

FMT_MSA.3.1/GP The TSF shall enforce the **ELF Loading information flow control SFP and Data & Key Loading information flow control SFP** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/GP The TSF shall allow the **None**[94] to specify alternative initial values to override the default values when an object or information is created.

Application Note:
- This SFR refines and replaces FMT_MSA.3/CM of [PP-JCS]. It is extended to cover the Data and Key loading Policy.
- The authorized identified roles could be off-card or on-card entities as defined in FMT_SMR.1/GP.

FDP_ACC.1/OS-UPDATE Subset access control

FDP_ACC.1.1/OS-UPDATE The TSF shall enforce the **OS Update Access Control Policy** on **the following list of subjects, objects, and operations**:
- **Subjects: S.OS-DEVELOPER is the representative of the OS Developer within the TOE, being responsible for signature verification and decryption of the additional code, before Loading, Installation, ~~and~~ Activation and none**[95] **are authorized.**
- **Objects: additional code and associated cryptographic signature**
- **Operations: loading, installation, and activation of additional code**

[94] [assignment: authorized identified roles]
[95] [assignment: list of other subjects covered by the SFP]

FDP_ACF.1/OS-UPDATE	Security attribute based access control

FDP_ACF.1.1/OS-UPDATE The TSF shall enforce the **OS Update Access Control Policy** to objects based on the following Security Attributes:

- **The additional code cryptographic signature verification status**
- **The Identification Data verification status (between the Initial TOE and the additional code)**

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

- **The verification of the additional code cryptographic signature (using D.OS-UPDATE_SGNVER-KEY) by S.OS-DEVELOPER is successful.**
- **The decryption of the additional code prior installation (using D.OS-UPDATE_DEC-KEY) by S.OS-DEVELOPER is successful.**
- **The comparison between the identification data of both the Initial TOE and the additional code demonstrates that the OS Update operation can be performed.**
- **none**[96]

FDP_ACF.1.3/OS-UPDATE The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: **none**[97].

FDP_ACF.1.4/OS-UPDATE The TSF shall explicitly deny access of subjects to objects based on the following additional rules: **none**[98].

Application Note:

- Identification data verification is necessary to ensure that the received additional code is actually targeting the TOE and that its version is compatible with the TOE version.
- Confidentiality protection must be enforced when the additional code is transmitted to the TOE for loading (See OE.OS-UPDATE-ENCRYPTION). Confidentiality protection is achieved through direct encryption of the additional code.

FMT_MSA.3/OS-UPDATE	Security attribute initialization

FMT_MSA.3.1/OS-UPDATE The TSF shall enforce the **OS Update Access Control Policy** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/OS-UPDATE The TSF shall allow the **OS Developer** to specify alternative initial values to override the default values when an object or information is created.

Application Note: the additional code signature verification status must be set to "Fail" by default. This prevents installation of any additional code until the additional code signature is successfully verified by the TOE.

FMT_SMR.1/OS-UPDATE	Security roles

FMT_SMR.1.1/OS-UPDATE The TSF shall maintain the roles **OS Developer, Issuer**.

[96] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[97] [assignment: rules, based on security attributes, that explicitly authorize access of subjects to objects]
[98] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]

FMT_SMR.1.2/OS-UPDATE The TSF shall be able to associate users with roles.

FMT_SMF.1/OS-UPDATE	Specification of Management Functions

FMT_SMF.1.1/OS-UPDATE The TSF shall be capable of performing the following management functions: **activation of additional code**.

Application Note: once verified and installed, additional code needs to be activated to become effective.

FIA_ATD.1/OS-UPDATE	User attribute definition

FIA_ATD.1.1/OS-UPDATE The TSF shall maintain the following list of security attributes belonging to individual users: **additional code ID for each activated additional code**.

Refinement: "Individual users" stands for additional code.

FTP_TRP.1/OS-UPDATE	Trusted Path

FTP_TRP.1.1/OS-UPDATE The TSF shall provide a communication path between itself and **remote** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from **none**[99].

FTP_TRP.1.2/OS-UPDATE The TSF shall permit **remote users** to initiate communication via the trusted path.

FTP_TRP.1.3/OS-UPDATE The TSF shall require the use of the trusted path for **the transfer of the additional code to the TOE**.

Application Note: during the transmission of the additional code to the TOE for loading, the confidentiality is ensured through direct encryption of the additional code, hence the 'none' selection in FTP_TRP.1.1/OS-UPDATE.

FCS_COP.1/OS-UPDATE-DEC	Cryptographic operation

FCS_COP.1.1/OS-UPDATE-DEC The TSF shall perform **Decryption of the additional code prior installation** in accordance with a specified cryptographic algorithm **AES in CBC mode with null IV**[100] and cryptographic key sizes **128 bits**[101] that meet the following: **FIPS 197**[102].

FCS_COP.1/OS-UPDATE-VER	Cryptographic operation

FCS_COP.1.1/OS-UPDATE-VER The TSF shall perform **digital signature verification of the additional code to be loaded** in accordance with a specified cryptographic algorithm **AES-CMAC**[103] and cryptographic key sizes **128 bits**[104] that meet the following: **FIPS 197 and SP800-38B**[105].

[99] [selection: disclosure, none]
[100] [assignment: cryptographic algorithm]
[101] [assignment: cryptographic key sizes]
[102] [assignment: list of standards]
[103] [assignment: cryptographic algorithm]
[104] [assignment: cryptographic key sizes]
[105] [assignment: list of standards]

FPT_FLS.1/OS-UPDATE	Failure with preservation of secure state

FPT_FLS.1.1/OS-UPDATE The TSF shall preserve a secure state when the following types of failures occur: **interruption or incident, which prevents the forming of the Updated TOE**.

Application Note:

- The OS Update operation must either be successful or fail securely. There are 3 steps in an OS Update operation:
  - step 1: loading
  - step 2: activation
  - step 3: update of TOE identification data

  Steps 2 and 3 are performed atomically, so that the TOE active code and identification data always remain consistent.
- If a failure (interruption or incident) occurs during step 1 (loading), then the TOE remains in its initial state (no update, neither of code nor of the TOE identification data).
- If a failure (interruption or incident) occurs during the atomic sequence step 2 / step 3 (activation / update of TOE identification data), then the enforced behavior depends on the nature of the update:
  - For java code updates, the TOE remains in its initial state and the OS Update operation is aborted.
  - For native code updates, the TOE does some retries to complete the atomic sequence step 2 / step three (activation / update of TOE identification data) until it is successful.
  - In any case, only two possible secure states are possible at any given time:
    - Either activation is not done and the TOE identification data is not updated (i.e. initial state)
    - Alternatively, the atomic sequence completes successfully, i.e. the OS update is activated and the TOE identification data is updated accordingly.

7.2.8 Underlying platform IC Security Functional Requirements

FAU_SAS.1 Audit Storage

FAU_SAS.1.1 The TSF shall provide **[assignment: the test process before TOE Delivery]** with the capability to store **[selection: the Initialisation Data, Pre-personalisation Data, [assignment: none]]** in the **[assignment: ST54L OST ROM]**.

FPT_RCV.3/OS	Automated recovery without undue loss

FPT_RCV.3.1/OS When automated recovery from **none, see application note below**[106] is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT_RCV.3.2/OS For **execution access to a memory zone reserved for TSF data, writing access to a memory zone reserved for TSF's code, and any segmentation fault performed by a Java Card applet**[107] the TSF shall ensure the return of the TOE to a secure state using automated procedures.

[106] [assignment: list of failures/service discontinuities during card content management operations]
[107] [assignment: list of failures/service discontinuities during card content management operations]

FPT_RCV.3.3/OS the functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding

- **the contents of Java Card static fields, instance fields, and array positions that fall under the scope of an open transaction;**
- **the Java Card objects that were allocated into the scope of an open transaction;**
- **the contents of Java Card transient objects;**
- **any possible Executable Load File being loaded when the failure occurred**[108]

For loss of TSF data or objects under the control of the TSF.

FPT_RCV.3.4/OS The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.

Application note: there is no maintenance mode implemented within the TOE. Recovery is always enforced automatically as stated in FPT_RCV.3.2/OS.

FPT_RCV.4/OS	**Function recovery**

FPT_RCV.4.1/OS The TSF shall ensure that **reading from and writing to static and objects' fields interrupted by power loss**[109] have the property that the function either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

[108] [assignment: quantification]
[109] [assignment: list of functions and failure scenarios]

7.3 Security Functional Requirements Rationale

7.3.1 SFRs for eUICC rationale

The security functional requirements rationale is the same than the ones present in section 6.3 from [PP-eUICC].

7.3.2 SFRs for Runtime Environment rationale

The security functional requirements rationale for objectives O.RE is extracted from [PP-JCS] and [PP-GP] and adapted depending on the implementation and the included SFRs and its iterations.*

The next table shows the objectives related to [PP-eUICC] runtime environment and its translation according to [PP-eUICC] application notes for OE.RE objectives. The security functional requirements rationale of O.RE* will be the same than the rationale for the objectives translated from JavaCard PP [PP-JCS] and are not repeated here. In case of O.CARD-MANAGEMENT, the Security Functional Requirements rationale is extracted from [PP-GP].*

RE objectives	Translation from JavaCard PP
O.RE.PPE-PPI	O.INSTALL, O.DELETION, O.LOAD, O.CARD-MANAGEMENT
O.RE.SECURE-COMM	O.SCP.RECOVERY, OE.SCP.SUPPORT, O.CARD-MANAGEMENT, O.SID, O.OPERATE, O.FIREWALL, O.GLOBAL_ARRAYS_CONFID, O.GLOBAL_ARRAYS_INTEG, O.ALARM, O.TRANSACTION, O.CIPHER, O.RNG, O.PIN-MNGT, O.KEY-MNGT, O.REALLOCATION
O.RE.API	O.CARD-MANAGEMENT, O.NATIVE, OE.SCP.RECOVERY, OE.SCP.SUPPORT, O.SID, O.OPERATE, O.FIREWALL, O.ALARM
O.RE.DATA-CONFIDENTIALITY	OE.SCP.RECOVERY, OE.SCP.SUPPORT, O.CARD-MANAGEMENT, O.SID, O.OPERATE, O.FIREWALL, O.GLOBAL_ARRAYS_CONFID, O.ALARM, O.TRANSACTION, O.CIPHER, O.RNG, O.PIN-MNGT, O.KEY-MNGT, O.REALLOCATION
O.RE.DATA-INTEGRITY	OE.SCP.RECOVERY, OE.SCP.SUPPORT, O.CARD-MANAGEMENT, O.SID, O.OPERATE, O.FIREWALL, O.GLOBAL_ARRAYS_INTEG, O.ALARM, O.TRANSACTION, O.CIPHER, O.RNG, O.PIN-MNGT, O.KEY-MNGT, O.REALLOCATION, O.LOAD, O.NATIVE
O.RE.IDENTITY	OE.SCP.RECOVERY and OE.SCP.SUPPORT, O.FIREWALL, O.SID, O.INSTALL, O.OPERATE, O.GLOBAL_ARRAYS_CONFID, O.GLOBAL_ARRAYS_INTEG, O.CARD-MANAGEMENT
O.RE.CODE-EXE	O.FIREWALL, O.REMOTE, O.NATIVE
O.SECURE_LOAD_ACODE	FDP_ACC.1/OS-UPDATE, FDP_ACF.1/OS-UPDATE, FMT_MSA.3/OS-UPDATE, FMT_SMR.1/OS-UPDATE, FMT_SMF.1/OS-UPDATE, FCS_COP.1/OS-UPDATE-VER
O.SECURE_AC_ACTIVATION	FDP_ACC.1/OS-UPDATE, FDP_ACF.1/OS-UPDATE, FMT_MSA.3/OS-UPDATE, FMT_SMR.1/OS-UPDATE, FMT_SMF.1/OS-UPDATE, FPT_FLS.1/OS-UPDATE
O.TOE_IDENTIFICATION	FDP_ACC.1/OS-UPDATE, FDP_ACF.1/OS-UPDATE, FIA_ATD.1/OS-UPDATE, FMT_MSA.3/OS-UPDATE, FMT_SMR.1/OS-UPDATE, FMT_SMF.1/OS-UPDATE
O.CONFID-OS-UPDATE.LOAD	FDP_ACC.1/OS-UPDATE, FDP_ACF.1/OS-UPDATE, FMT_MSA.3/OS-UPDATE, FMT_SMR.1/OS-UPDATE, FMT_SMF.1/OS-UPDATE, FTP_TRP.1/OS-UPDATE, FCS_COP.1/OS-UPDATE-DEC

Table 19 - Runtime environment objectives conversion for SFR rationale.

Note that OE.SCP.RECOVERY and OE.SCP.SUPPORT from [PP-JCS] are equivalent to OE.IC.RECOVERY and OE.IC.SUPPORT from [PP-eUICC] converted to O.IC.RECOVERY and O.IC.SUPPORT in current Security Target. See next section for the rationale.

Note that For O.RE.IDENTITY, this objective is translated from OE.RE.IDENTITY to cover the following threats: T.UNAUTHORIZED-IDENTITY-MNG.

7.3.3 SFRs for underlying platform IC rationale

O.IC.PROOF_OF_IDENTITY coverage: the IC is a part of the TOE supporting TSFs of the upper layer of the TOE, especially for identification data storage as dealt with FAU_SAS.1.

O.IC.RECOVERY coverage: the IC is a part of the TOE supporting TSFs of the upper layer of the TOE, especially for recovery operations as dealt with in FPT_RCV.3/OS.

O.IC.SUPPORT coverage: the IC is a part of the TOE supporting TSFs of the upper layer of the TOE, especially for recovery operations as dealt with in FPT_RCV.4/OS.

7.3.4 SFRs dependency rationale

SFR	CC dependencies	Satisfied dependencies
FIA_UID.1/EXT	No Dependencies	
FIA_UAU.1/EXT	(FIA_UID.1)	FIA_UID.1/EXT
FIA_USB.1/EXT	(FIA_ATD.1)	FIA_ATD.1
FIA_UAU.4/EXT	No Dependencies	
FIA_UID.1/MNO-SD	No Dependencies	
FIA_USB.1/MNO-SD	(FIA_ATD.1)	FIA_ATD.1
FIA_ATD.1	No Dependencies	
FIA_API.1	No Dependencies	
FDP_IFC.1/SCP	(FDP_IFF.1)	FDP_IFF.1/SCP
FDP_IFF.1/SCP	(FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.1/SCP, FMT_MSA.3
FTP_ITC.1/SCP	No Dependencies	
FDP_ITC.2/SCP	(FDP_ACC.1 or FDP_IFC.1) and (FPT_TDC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_IFC.1/SCP, FTP_ITC.1/SCP, FPT_TDC.1/SCP
FPT_TDC.1/SCP	No Dependencies	
FDP_UCT.1/SCP	(FDP_ACC.1 or FDP_IFC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_IFC.1/SCP, FTP_ITC.1/SCP
FDP_UIT.1/SCP	(FDP_ACC.1 or FDP_IFC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_IFC.1/SCP, FTP_ITC.1/SCP

FCS_CKM.1/SCP-SM	(FCS_CKM.2 or FCS_COP.1) and (FCS_CKM.4)	FCS_COP.1/ECKA_EG, FCS_COP.1/GP-SCP, FCS_CKM.4/SCP-SM
FCS_CKM.2/SCP-MNO	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	FDP_ITC.2/SCP, FCS_CKM.4/SCP-MNO
FCS_CKM.4/SCP-SM	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2)	FDP_ITC.2/SCP, FCS_CKM.1/SCP-SM
FCS_CKM.4/SCP-MNO	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2)	FDP_ITC.2/SCP, FCS_CKM.1/SCP-SM
FDP_ACC.1/ISDR	(FDP_ACF.1)	FDP_ACF.1/ISDR
FDP_ACF.1/ISDR	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.1/ISDR, FMT_MSA.3
FDP_ACC.1/ECASD	(FDP_ACF.1)	FDP_ACF.1/ECASD
FDP_ACF.1/ECASD	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.1/ECASD, FMT_MSA.3
FDP_IFC.1/Platform_services	(FDP_IFF.1)	FDP_IFF.1/Platform_services
FDP_IFF.1/Platform_services	(FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.1/Platform_services, FMT_MSA.3
FPT_FLS.1/Platform_services	No Dependencies	
FCS_RNG.1	No Dependencies	
FPT_EMS.1	No Dependencies	
FDP_SDI.1	No Dependencies	
FDP_RIP.1	No Dependencies	
FPT_FLS.1	No Dependencies	
FMT_MSA.1/PLATFORM_DATA	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.1/ISDR, FMT_SMF.1, FMT_SMR.1
FMT_MSA.1/PPR	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.1/ISDR, FMT_SMF.1, FMT_SMR.1
FMT_MSA.1/CERT_KEYS	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.1/ISDR, FMT_SMF.1, FMT_SMR.1
FMT_SMF.1	No Dependencies	
FMT_SMR.1	(FIA_UID.1)	FIA_UID.1/EXT, FIA_UID.1/MNO-SD
FMT_MSA.1/RAT	(FDP_ACC.1 or FDP_IFC.1) and	FDP_ACC.1/ISDR, FMT_SMF.1, FMT_SMR.1

	(FMT_SMF.1) and (FMT_SMR.1)	
FMT_MSA.3	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/PLATFORM_DATA, FMT_MSA.1/PPR, FMT_MSA.1/CERT_KEYS, FMT_SMR.1, FMT_MSA.1/RAT
FCS_COP.1/Mobile_network	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	FDP_ITC.2/SCP, FCS_CKM.4/Mobile_network
FCS_CKM.2/Mobile_network	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	FMT_MSA.1/PLATFORM_DATA, FMT_MSA.1/PPR, FMT_MSA.1/CERT_KEYS, FMT_SMR.1, FMT_MSA.1/RAT
FCS_CKM.4/Mobile_network	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2)	FDP_ITC.2/SCP, FCS_CKM.4/Mobile_network
FDP_ACC.2/FIREWALL	(FDP_ACF.1)	FDP_ACF.1/FIREWALL
FDP_ACF.1/FIREWALL	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.2/FIREWALL FMT_MSA.3/FIREWALL
FDP_IFC.1/JCVM	(FDP_IFF.1)	FDP_IFF.1/JCVM
FDP_IFF.1/JCVM	(FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.1/JCVM FMT_MSA.3/JCVM
FDP_RIP.1/OBJECTS	No Dependencies	
FMT_MSA.1/JCRE	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.2/FIREWALL **See rationale** FMT_SMR.1/JC
FMT_MSA.1/JCVM	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.2/FIREWALL FDP_IFC.1/JCVM FMT_SMF.1/CM FMT_SMR.1/JC
FMT_MSA.2/FIREWALL_JCVM	(FDP_ACC.1 or FDP_IFC.1) and (FMT_MSA.1) and (FMT_SMR.1)	FDP_ACC.2/FIREWALL FDP_IFC.1/JCVM FMT_MSA.1/JCRE FMT_MSA.1/JCVM FMT_SMR.1/JC
FMT_MSA.3/FIREWALL	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/JCRE FMT_MSA.1/JCVM FMT_SMR.1/JC
FMT_MSA.3/JCVM	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/JCVM FMT_SMR.1/JC
FMT_SMF.1/JC	No Dependencies	
FMT_SMR.1/JC	(FIA_UID.1)	FIA_UID.2/AID
FCS_CKM.1/ECDSA FCS_CKM.1/GP-SCP	(FCS_CKM.2 or FCS_COP.1) and (FCS_CKM.4)	FCS_COP.1/RSA_SIGN FCS_COP.1/RSA_CIPHER FCS_COP.1/ECDSA_SIGN FCS_COP.1/ECDH FCS_COP.1/GP-SCP FCS_CKM.4
FCS_CKM.4	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2)	

FCS_COP.1/TDES_MAC FCS_COP.1/AES_MAC FCS_COP.1/ECDH FCS_COP.1/CRC FCS_COP.1/ECDSA_SIGN FCS_COP.1/ECKA_EG FCS_COP.1/GP-SCP FCS_COP.1/TDES_CIPHER FCS_COP.1/AES_CIPHER FCS_COP.1/RSA_SIGN FCS_COP.1/RSA_CIPHER FCS_COP.1/Hash FCS_COP.1/HMAC	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	FCS_CKM.1/ECDSA FCS_CKM.1/GP-SCP FCS_CKM.4 **See rationale**
FDP_RIP.1/ABORT	No Dependencies	
FDP_RIP.1/APDU	No Dependencies	
FDP_RIP.1/bArray	No Dependencies	
FDP_RIP.1/GlobalArray	No Dependencies	
FDP_RIP.1/KEYS	No Dependencies	
FDP_RIP.1/TRANSIENT	No Dependencies	
FDP_ROL.1/FIREWALL	(FDP_ACC.1 or FDP_IFC.1)	FDP_ACC.2/FIREWALL FDP_IFC.1/JCVM
FAU_ARP.1	(FAU_SAA.1)	**See rationale**
FDP_SDI.2/DATA	No Dependencies	
FPR_UNO.1	No Dependencies	
FPR_FLS.1	No Dependencies	
FPT_TDC.1	No Dependencies	
FIA_ATD.1/AID	No Dependencies	
FIA_UID.2/AID	No Dependencies	
FIA_USB.1/AID	(FIA_ATD.1)	FIA_USB.1/AID
FMT_MTD.1/JCRE	(FMT_SMF.1) and (FMT_SMR.1)	FMT_SMF.1/CM FMT_SMR.1/JC
FMT_MTD.3/JCRE	(FMT_MTD.1)	FMT_MTD.1/JCRE
FDP_ACC.2/ADEL	(FDP_ACF.1)	FDP_ACF.1/ADEL
FDP_ACF.1/ADEL	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.2/ADEL FMT_MSA.3/ADEL
FDP_RIP.1/ADEL	No Dependencies	
FMT_MSA.1/ADEL	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.2/ADEL FMT_SMF.1/ADEL FMT_SMR.1/ADEL
FMT_MSA.3/ADEL	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/ADEL FMT_SMR.1/ADEL
FMT_SMF.1/ADEL	No Dependencies	
FMT_SMR.1/ADEL	(FIA_UID.1)	**See rationale**
FPT_FLS.1/ADEL	No Dependencies	
FDP_RIP.1/ODEL	No Dependencies	
FPT_FLS.1/ODEL	No Dependencies	
FPT_FLS.1/GP	No Dependencies	
FDP_ROL.1/GP	(FDP_ACC.1 or FDP_IFC.1)	FDP_IFC.2/GP-ELF FDP_IFC.2/GP-KL
FCO_NRO.2/GP	(FIA_UID.1)	FIA_UID.1/GP
FMT_SMR.1/GP	(FIA_UID.1)	FIA_UID.1/GP
FMT_SMF.1/GP	No Dependencies	
FDP_ITC.2/GP-ELF	(FDP_ACC.1 or FDP_IFC.1) and	FDP_IFC.2/GP-ELF FPT_TDC.1/GP

	(FPT_TDC.1) and (FTP_ITC.1 or FTP_TRP.1)	FTP_ITC.1/GP
FDP_ITC.2/GP-KL	(FDP_ACC.1 or FDP_IFC.1) and (FPT_TDC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_IFC.2/GP-KL FPT_TDC.1/GP FTP_ITC.1/GP
FPT_RCV.3/GP	(AGD_OPE.1)	AGD_OPE.1
FDP_IFC.2/GP-ELF	(FDP_IFF.1)	FDP_IFF.1/GP-ELF
FDP_IFF.1/GP-ELF	(FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.2/GP-ELF FMT_MSA.3/GP
FIA_UID.1/GP	No Dependencies	
FIA_AFL.1/GP	(FIA_UAU.1)	FIA_UAU.1/GP
FIA_UAU.1/GP	(FIA_UID.1)	FIA_UID.1/GP
FIA_UAU.4/GP	No Dependencies	
FDP_UIT.1/GP	(FDP_ACC.1 or FDP_IFC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_IFC.2/GP-ELF FDP_IFC.2/GP-KL FTP_ITC.1/GP
FDP_UCT.1/GP	(FTP_ITC.1 or FTP_TRP.1) and (FDP_ACC.1 or FDP_IFC.1)	FDP_IFC.2/GP-ELF FDP_IFC.2/GP-KL FTP_ITC.1/GP
FTP_ITC.1/GP	No Dependencies	
FPR_UNO.1/GP	No Dependencies	
FPT_TDC.1/GP	No Dependencies	
FDP_IFC.2/GP-KL	(FDP_IFF.1)	FDP_IFF.1/GP-KL
FDP_IFF.1/GP-KL	(FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.2/GP-KL FMT_MSA.3/GP
FMT_MSA.1/GP	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_IFC.2/GP-ELF FDP_IFC.2/GP-KL FMT_SMR.1/GP FMT_SMF.1/GP
FMT_MSA.3/GP	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/GP FMT_SMR.1/GP
FDP_ACC.1/OS-UPDATE	(FDP_ACF.1)	FDP_ACF.1/OS-UPDATE
FDP_ACF.1/OS-UPDATE	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.1/OS-UPDATE FMT_MSA.3/OS-UPDATE
FMT_MSA.3/OS-UPDATE	(FMT_MSA.1) and (FMT_SMR.1)	FMT_SMR.1/OS-UPDATE **See rationale**
FMT_SMR.1/OS-UPDATE	(FIA_UID.1)	FIA_UID.1/GP
FMT_SMF.1/OS-UPDATE	No Dependencies	
FIA_ATD.1/OS-UPDATE	No Dependencies	
FTP_TRP.1/OS-UPDATE	No Dependencies	
FCS_COP.1/OS-UPDATE-DEC	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	FDP_ITC.2/GP-ELF FCS_CKM.4
FCS_COP.1/OS-UPDATE-VER	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	FDP_ITC.2/GP-ELF FCS_CKM.4
FPT_FLS.1/OS-UPDATE	No dependencies	
FAU_SAS.1	No Dependencies	
FPT_RCV.3/OS	(AGD_OPE.1)	AGD_OPE.1
FPT_RCV.4/OS	No Dependencies	

Table 20 – SFRs dependency table

Rationale for the exclusion of dependencies:

- **The dependency FMT_SMF.1 of FMT_MSA.1/JCRE is unsupported.**

The dependency between FMT_MSA.1/JCRE and FMT_SMF.1 is not satisfied because no management functions are required for the Java Card RE.

- **The dependencies of FCS_COP.1/Hash are unsupported**

Hash operation does not require any key.

- **The dependencies of FCS_COP.1/CRC are unsupported**

CRC operations do not require any key.

- **The dependency FAU_SAA.1 of FAU_ARP.1 is unsupported**

The dependency of FAU_ARP.1 on FAU_SAA.1 assumes that a "potential security violation" generates an audit event. On the contrary, the events listed in FAU_ARP.1 are self-contained (arithmetic exception, ill-formed bytecodes, access failure) and ask for a straightforward reaction of the TSFs on their occurrence at runtime. The JCVM or other components of the TOE detect these events during their usual working order. Thus, there is no mandatory audit recording in this ST.

- **The dependency FIA_UID.1 of FMT_SMR.1/ADEL is unsupported**

This ST does not require the identification of the "deletion manager" since it can be considered as part of the TSF.

- **The dependency FMT_MSA.1 of FMT_MSA.3/OS-UPDATE is unsupported.**

No history information has to be kept by the TOE.

8 TOE SUMMARY SPECIFICATION

The TOE implements the SFRs in accordance to the GSMA specifications, sufficiently hardened to counter attackers at AVA_VAN.5 level.

The TOE is equipped with following Security Features to meet the security functional requirements

8.1 eUICC security functions

8.1.1 GSMA.ProfileManagement

This security function implements the controls related to profiles management as defined by **[SGP.22]** and **[EUPP]**, encompassing the following operations:

- Profile downloading
- Profile elements installation
- Profile deletion
- Profile enable and disable

It also supports everything related to profile data isolation.

8.1.2 GSMA.ECASD

This security function handles the Embedded UICC Controlling Authority Security Domain (ECASD) management as defined by **[SGP.22].** The ECASD is responsible for secure storage of credentials required to support the required Security Domains on the eUICC.

ECASD installation, provisioning, eUICC authentication and credentials management are covered.

8.1.3 GSMA.ISDR

This security function handles the ISD-R management as defined by **[SGP.22]**. The ISD-R is responsible for the creation of new ISD-Ps and lifecycle management of all ISD-Ps.

ISD-R installation, provisioning, credentials and content management are covered.

8.1.4 GSMA.ISDP

This security function handles the ISD-P management as defined by **[SGP.22]**. The ISD-P is the on-card representative of the SM-DP+ and is a secure container (Security Domain) for the hosting of a Profile. The ISD-P is used for the Profile download and installation in collaboration with the Profile Package Interpreter for the decoding/interpretation of the received Profile Package.

ISD-R installation, provisioning, deletion, credentials and content management are covered.

8.1.5 GSMA.PPR

This security function implements Profile Policy Rules management as defined by **[SGP.22]**. The PPRs are defined by the Profile Owners and set by the SM-DP+ in the Profile Metadata. Upon downloading a profile with defined PPR, eUICC is required to follow these defined rules.

Secure management and processing of the PPRs are covered.

8.2 Runtime Environment security functions

8.2.1 GP.CardContentManagement

This security function provides the capability and a dedicated flow control for the loading, installation, extradition, registry update, selection and removal of card content and especially executable files and application instances. Such features are offered to the Card Issuer and its business partners, allowing the Card Issuer to delegate card content management to an Application Provider according to privileges assigned to the various security domains on the card. It supports Delegated management (DM), Authorized management (AM) and it can use DAP or Mandated DAP verification and generation of Reception token. It also checks that only the card management commands specified and allowed at each state of the smart card's life cycle are accepted, and ill-formed ones are rejected with an appropriate error response

8.2.2 GP.KeyLoading

This security function provides the capability and a dedicated flow control for the loading of keys and other sensitive data using the GlobalPlatform STORE DATA and PUT KEY APDUs, or by using GlobalPlatform APIs for loading and storing data and keys.

8.2.3 GP.SecurityDomain

This security function provides security domain management, as SD creation, SD selection, SD privileges setting and SD deletion in SD hierarchy. It provides means to associate or extradite an application to a security domain in order to provide services (as secure channel) to the dedicated application without sharing the related keys stored in SD. It also provides Keyset Management in SD, with Key Set creation, Key set deletion, key importation, replacement, or deletion in Key Set. Security Domains are privileged Applications as defined in [GPCS] § 7, holding cryptographic keys to be used to support Secure Channel Protocol operations and/or to authorize card content management functions. There are different types of security domain with dedicated privileges and associated operations: ISD Security domain, Supplementary Security domains, and Controlling Authority Security domains.

ISD Security domain as defined in [GPCS] §7.1.1, is the mandatory Security Domain, implicitly selected if the Application implicitly selectable on the same logical channel of the same card I/O interface is removed. It inherits of the Final Application privilege if the Application with that privilege is removed.

Supplementary Security Domains are privileged Applications with dedicated privileges:

- Token Verification Privilege as described in [GPCS] §9.1.3.1
- Authorized Management Privilege as described in [GPCS] §9.1.3.2
- Delegated Management Privilege as described in [GPCS] §9.1.3.3
- Global Delete Privilege as described in [GPCS] §9.1.3.4
- Global Lock Privilege as described in [GPCS] §9.1.3.5
- Receipt Generation Privilege as described in [GPCS] §9.1.3.6
- Ciphered Load File Data Block Privilege as described in [GPCS] §9.1.3.7

Controlling Authority Security Domain is a supplementary Security Domain dedicated to the Controlling Authority with dedicated privileges. It contains Security Domains cryptographic keys needed to confidentially personalize an initial set of Secure Channel Keys of an APSD.

8.2.4 GP.SecureChannel

This security function provides a secure communication channel between a card and an off-card entity during an Application Session according to [GPCS], [Amd B], [Amd D], [Amd F], [TS 102.225] and [TS 102.226]. It provides an APDU flow control using the Command security level check according to Card Life cycle and type of APDU.

A Secure Channel Session is divided into three sequential phases:

- Secure Channel Initiation when the on-card Application and the off-card entity have exchanged sufficient information enabling them to perform the required cryptographic functions. The Secure Channel Session initiation always includes (at least) the authentication of the off-card entity by the on-card Application; performing also the setting of the Command security level used for the session.
- Secure Channel Operation when the on-card Application and the off-card entity exchange data within the cryptographic protection of the Secure Channel Session. The Secure Channel services offered may vary from one Secure Channel Protocol to the other;
- Secure Channel Termination when either the on-card Application or the off-card entity determines that no further communication is required or allowed via an established Secure Channel Session.

The Secure Channel provides the following services:

- Entity authentication in which the card or the off-card entity proves its authenticity to the other entity through a cryptographic exchange, based on session key generation and a dedicated flow control; For SCP80, envelope APDU shall contain secured packet structure defined in [TS 102.225] §5 and Anti-replay mechanism is proposed optionally using a counter defined in [TS 102.225] §5.1.4;
- Integrity and authentication in which the receiving entity (the card or off-card entity) ensures that the data being received from the sending entity (respectively the off-card entity or card) actually came from an authenticated entity in the correct sequence and has not been altered;
- Confidentiality in which data being transmitted from the sending entity (the off-card entity or card) to the receiving entity (respectively the card or off-card entity) is not viewable by an unauthenticated entity.

The TOE supports the following Secure Channel Protocols: SCP02, SCP03, SCP11, SCP80 and SCP81.

8.2.5 GP.GPRegistry

This security function provides management and access to the GlobalPlatform Registry used for:

- Store card management information;
- Store relevant application management information (e.g., AID, associated Security Domain and Privileges);
- Support card resource management data;
- Store Application Life Cycle information;
- Store card Life Cycle information;
- Track any counters associated with logs.

The content of the GlobalPlatform Registry may be accessed by administrative commands or by applet using a dedicated GlobalPlatform API.
Only secure values are accepted for the information stored in the GlobalPlatform registry (including Life Cycle states, Security Levels and Privileges).

8.2.6 GP.OS-UPDATE

The TOE implements an OS Update capability by means of the GemActivate proprietary mechanism, allowing the **Connected eSE 5.3.4 v1.2** Platforms to be updated post-issuance (during phase 7 of the card life-cycle). OS updates are performed through the loading, installation and activation of related ELFs, fulfilling the same rules as for any other ELF. DAP verification (AES128 CMAC) is mandatory for ELFs containing OS updates, ensuring the authenticity and integrity protection of the code update, and the content of the ELF is directly encrypted (AES128 in CBC mode) with a dedicated encryption key, ensuring the confidentiality protection. Note that both the DAP signature verification key and the encryption key are GemActivate keys, meaning that OS updates can only be issued and decrypted by Thales. Verification of TOE identification data is also enforced before allowing any OS update. The whole

OS update operation is done through an atomic process, ensuring the permanent consistency between the **Connected eSE 5.3.4 v1.2** Platform active code and its identification data.

A secure state is preserved in case of failure during the OS update process. More precisely:

- There are 3 steps in an OS Update operation:
  - step 1: loading
  - step 2: activation
  - step 3: update of TOE identification data

  Steps 2 and 3 are performed atomically, so that the TOE active code and identification data always remain consistent.
- If a failure (interruption or incident) occurs during step 1 (loading), then the TOE remains in its initial state (no update, neither of code nor of the TOE identification data).
- If a failure (interruption or incident) occurs during the atomic sequence step 2 / step 3 (activation / update of TOE identification data), then the enforced behavior depends on the nature of the update:
  - For java code updates, the TOE remains in its initial state and the OS Update operation is aborted.
  - For native code updates, the TOE does some retries to complete the atomic sequence step 2 / step three (activation / update of TOE identification data) until it is successful.
  - In any case, only two possible secure states are possible at any given time:
    - Either activation is not done and the TOE identification data is not updated (i.e. initial state)
    - Alternatively, the atomic sequence completes successfully, i.e. the OS update is activated and the TOE identification data is updated accordingly.

8.2.7 JCS.APDUBuffer

The security function maintains a byte array buffer accessible from any applet context. This buffer is used to transfer incoming APDU header and data bytes as well as outgoing data according to [JCAPI3]. The APDU class API is designed to be transport protocol independent T=0, T=1, T=CL (as defined in ISO 7816-3).
Application note: ADPU buffer is a JCRE temporary entry point object where no associated reference can be stored in a variable or an array component.

8.2.8 JCS.ByteCodeExecution

This security function handles applet bytecode execution according to the rules defined in [JCVM3]. The JCVM execution may be summarized in JCVM interpreter start-up, bytecode execution and JCVM interpreter loop. The applet bytecode execution consists in:

- fetching the next bytecode to execute according to the applet's code flow control,
- decoding the next bytecode,
- Executing the fetched bytecode.

The JCVM manages several types of objects, such as persistent objects, transient objects, persistent arrays (boolean, byte, short, int or reference), transient arrays (boolean, byte, short, int or reference) and static field images. For each type of object, different types of control are performed.

8.2.9 JCS.Firewall

This security function enforces a Firewall access control policy and a JCVM information flow control policy at runtime. It defines how accessing the following items: Static Class Fields, Array Objects, Class Instance Object Fields, Class Instance Object Methods, Standard Interface Methods, Shareable Interface Methods, Classes, Standard Interfaces, Shareable Interfaces, Array Object Methods.
Based on security attributes (Sharing, Context, Lifetime), it performs access control to object fields between objects and throws security exception when access is denied. Thus, it enforces applet

isolation located in different packages and controls the access to global data containers shared by all applet instances.
The JCRE shall allocate and manage a context for each Java API package containing applets. The JCRE maintains for its own context a special system privilege so that it can perform operations that are denied to contexts of applets.

8.2.10 JCS.Package

This security function manages packages. A package is a structural item defined for naming, loading, storing, execution context definition. There are rules for package identification, for structure check and access rules definition. If inconsistent items are found during checks, an error message is sent.

8.2.11 JCS.CryptoAPI

This security function offers the following cryptographic services to applets through the JavaCard API:

- Generation of random numbers as defined in [JCAPI3] to be used for key values or challenges during external exchanges. The Random Number Generator (RNG) is hybrid deterministic and conformant to [AIS31] DRG.4, providing enhanced backward secrecy & enhanced forward secrecy. It passes [AIS31] test procedure A.
- Computation of checksum CRC16 and CRC32 conformant with ISO3309, as defined in [JCAPI3] Checksum class. Both ALG_ISO3309_CRC16 and ALG_ISO3309_CRC32 are supported.
- Encryption and decryption using TDES algorithm as defined in [JCAPI3] Cipher class. Both TDES 2-keys (112 bits key length) and TDES 3-keys (168 bits key length) are supported.
- Generation of 4-byte or 8-byte MAC using TDES algorithm as defined in [JCAPI3] Signature class. Both TDES 2-keys (112 bits key length) and TDES 3-keys (168 bits key length) are supported.
- Encryption and decryption using AES (128, 192 or 256 bits key) algorithm as defined in [JCAPI3] Cipher class.
- Generation of 16-byte, 24-byte or 32-byte MAC using AES algorithm (128, 192 or 256 bits key) in CBC mode as defined in [JCAPI3] Signature class.
- Data hash computation as defined in [JCAPI3] MessageDigest class.
- HMAC computation as defined in [JCAPI3] Signature class.
- Encryption and decryption using RSA with Standard or CRT modes, as defined in [JCAPI3] Cipher class. All key lengths from 1024 to 2048 bits (by steps of 32 bits) are supported.
- Generation and verification of RSA signatures in Standard or CRT modes, as defined in [JCAPI3] Signature class. All key lengths from 1024 to 2048 bits (by steps of 32 bits) are supported.
- Generation and verification of ECDSA signatures as defined in [JCAPI3] Signature class. Elliptic curve cryptography over GF(p) is considered here, with P ranging from 160 to 521 bits.
- Secret key agreement according to the ECDH algorithm, as defined in [JCAPI3] KeyAgreement class.
- Secret key agreement according to the DH algorithm (ALG_DH_PLAIN), as defined in [JCAPI3] KeyAgreement class. RSA key sizes ranging from 1024 to 2048 bits (by steps of 32 bits) are supported.

These operations are performed in a way to avoid revealing the key values. If the applet specifies an algorithm that the platform does not support, the JCRE refuses to perform the cryptographic operation and generates an exception.

8.2.12 JCS.KeyManagement

This security function enforces key management for the different associated operations: key building and generation, key importation, key exportation, key masking and key destruction using the standard API defined in [JCAPI3].

- Key generation implemented through KeyBuilder and/or KeyPair classes : TDES key generation (112 or 168 bits), AES key generation (128, 192 or 256 bits), RSA Standard and RSA CRT Key Pair Generation (1024 to 2048 bits by steps of 32 bits), ECDSA Key Pair Generation (P ranging from 160 to 521 bits) and HMAC Key generation.
- Key importation and exportation is done using method protecting confidentiality and integrity of key.
- Key masking protects the confidentiality of cryptographic keys from being read out from the memory. It ensures the service of accessing and modifying them.
- Key destruction (implemented through the method clearKey() of the Key class) disables the use of a key both logically and physically. Reuse is only possible after erase.

8.2.13 JCS.OwnerPIN

This security function provides to applets a means to perform user identification and authentication with the OwnerPin class conformant to [JCAPI3].
It offers to create a PIN and store it securely in the persistent memory. It allows access to PIN value only to perform a secure comparison between a PIN stored in the persistent memory and a data received as parameter.
A method returns a positive result if a valid Pin has been presented during current session. If the PIN is not blocked and the comparison is successful, the validated flag is set to and the try counter is set to its maximum, otherwise the authentication fails and the associated try counter is decremented. When the validated flag is set, it is assumed that the user is authenticated.
When the try counter reaches zero, the PIN is blocked and the authentication is no more possible until the PIN is unblocked.

8.2.14 JCS.EraseResidualData

This security function ensures that sensitive data are locked upon the following operations as defined in [JCRE3]:

- Deletion of package and/or applications,
- Deletion of objects.

They are erased when space needs to be reused for allocation of new objects.
This security function also ensures that the sensitive temporary buffers (transient object, bArray object, Global Array object, APDU buffer, Cryptographic buffer) are securely cleared after their usage with respect to their life-cycle and interface as defined in [JCRE3], transient object at reset or allocation and persistent object are erased at allocation for new object.

8.2.15 JCS.OutOfLifeDataUndisclosure

This security function ensures that sensitive data are locked until postponed erasure on the following operations: Deletion of persistent and transient objects according to [JCRE3].

8.2.16 JCS.RunTimeExecution

This security function provides a secure run time environment conformant to [JCRE3] and deals with:

- Instance registration or deletion,
- Application selection,
- Applet opcode execution,
- JCAPI methods execution,
- Logical channel management,
- APDU flow control, dispatch and buffer management,
- JCRE memory and context management,
- JCRE reference deletion,
- JCRE access rights,
- JCRE throw exception,

- JCRE security reaction.

8.2.17 JCS.Exception

This security function manages throwing of an instance of Exception class in the following cases:

- a SecurityException when an illegal access to an object is detected,
- a SystemException with an error code describing the error condition,
- a CryptoException in case of algorithm error or illegal use,
- Any exception decided by the applet or the JCRE handled as temporary JCRE entry point object with associated JCAPI. It also offers a means to applet to handle exception and to JCRE to handle uncaught exception by applets.

8.2.18 OS.Atomicity

This security function performs write operations atomically on complex type or object in order to avoid incomplete update. Prior to be written, data is stored in an atomic back-up area. In case on writing interrupt, the only two possible values are initial value if writing is not started or final value if writing is started. At next start-up, the atomic back-up area is check to finalize interrupted writing.

8.2.19 OS.MemoryManagement

This security function allocates memory areas and performs access control on them to avoid unauthorized access. It manages circular writing to avoid instable memory state. It enforces memory recovery in case of error detection. It offers (when required) confidentiality services for data storage: Ciphering / Deciphering of Data in RAM or in FLASH, Scrambling / Unscrambling of Data in RAM or in FLASH.

8.3 TSS Rationale

The justification and overview of the mapping between security functional requirements (SFR) and the TOE's security functionality (SF) is given in section above.

8.3.1 eUICC SFRs coverage

Security Functional Requirement	Coverage by TSS Security Function(s)
FIA_UID.1/EXT	This SFR is covered by GSMA.ISDR
FIA_UAU.1/EXT	This SFR is covered by GSMA.ECASD and GP.SecureChannel
FIA_USB.1/EXT	This SFR is covered by GSMA.ECASD and GP.SecurityDomain
FIA_UAU.4/EXT	This SFR is covered by GSMA.ECASD and GP.SecureChannel
FIA_UID.1/MNO-SD	This SFR is covered by GP.SecurityDomain
FIA_USB.1/MNO-SD	This SFR is covered by GP.SecurityDomain, GSMA.ISDP, GSMA.ECASD
FIA_ATD.1	This SFR is covered by GP.SecurityDomain and GSMA.ECASD
FIA_API.1	This SFR is covered by GSMA.ECASD
FDP_IFC.1/SCP	This SFR is covered by GSMA.ProfileManagement
FDP_IFF.1/SCP	This SFR is covered by GSMA.ProfileManagement
FTP_ITC.1/SCP	This SFR is covered by GSMA.ProfileManagement
FDP_ITC.2/SCP	This SFR is covered by GSMA.ProfileManagement
FPT_TDC.1/SCP	This SFR is covered by GSMA.ProfileManagement
FDP_UCT.1/SCP	This SFR is covered by GSMA.ProfileManagement
FDP_UIT.1/SCP	This SFR is covered by GSMA.ProfileManagement
FCS_CKM.1/SCP-SM	This SFR is covered by GSMA.ProfileManagement and JCS.CryptoAPI for ECKA-EG
FCS_CKM.2/SCP-MNO	This SFR is covered by JCS.CryptoAPI
FCS_CKM.4/SCP-SM	This SFR is covered by JCS.KeyManagement
FCS_CKM.4/SCP-MNO	This SFR is covered by JCS.KeyManagement
FDP_ACC.1/ISDR	This SFR is covered by GSMA.ISDR
FDP_ACF.1/ISDR	This SFR is covered by GSMA.ISDR
FDP_ACC.1/ECASD	This SFR is covered by GSMA.ECASD
FDP_ACF.1/ECASD	This SFR is covered by GSMA.ECASD
FDP_IFC.1/Platform_services	This SFR is covered by GSMA.ProfileManagement
FDP_IFF.1/Platform_services	This SFR is covered by GSMA.ProfileManagement
FPT_FLS.1/Platform_services	This SFR is covered by GSMA.ProfileManagement
FCS_RNG.1	This SFR is covered by JCS.CryptoAPI providing AIS31 DRG.4 random number generation to applets.
FPT_EMS.1	This SFR is covered by JCS.CryptoAPI and JCS.KeyManagement
FDP_SDI.1	This SFR is covered by GSMA.ProfileManagement
FDP_RIP.1	This SFR is covered by GSMA.ProfileManagement
FPT_FLS.1	This SFR is covered by GSMA.ProfileManagement
FMT_MSA.1/PLATFORM_DATA	This SFR is covered by GSMA.ISDR
FMT_MSA.1/PPR	This SFR is covered by GSMA.PPR
FMT_MSA.1/CERT_KEYS	This SFR is covered by GSMA.ProfileManagement
FMT_SMF.1	This SFR is covered by GSMA.ProfileManagement, GSMA.ISDR, GSMA.ISDP, GSMA.ECASD, and GSMA.PPR
FMT_SMR.1	This SFR is covered by GSMA.ProfileManagement, GSMA.ISDR, GSMA.ISDP, GSMA.ECASD, and GSMA.PPR
FMT_MSA.1/RAT	This SFR is covered by GSMA.ISDR
FMT_MSA.3	This SFR is covered by GSMA.ISDR, GSMA.ISDP, GSMA.ECASD

Security Functional Requirement	Coverage by TSS Security Function(s)
FCS_COP.1/Mobile_network	This SFR is covered by JCS.CryptoAPI
FCS_CKM.2/Mobile_network	This SFR is covered by JCS.CryptoAPI
FCS_CKM.4/Mobile_network	This SFR is covered by JCS.KeyManagement

8.3.2 Runtime Environment SFRs coverage

Security Functional Requirement	Coverage by TSS Security Function(s)
FDP_ACC.2/FIREWALL	This SFR is covered by JCS.Firewall.
FDP_ACF.1/FIREWALL	This SFR is covered by JCS.Firewall.
FDP_IFC.1/JCVM	This SFR is covered by JCS.Firewall and JCS.APDUBuffer controlling unauthorized access or invalid storage of reference.
FDP_IFF.1/JCVM	This SFR is covered by JCS.Firewall.
FDP_RIP.1/OBJECTS	This SFR is covered by JCS.OutOfLifeDataUndisclosure (to avoid access to data prior erase) and JCS.EraseResidualData (to erase data).
FMT_MSA.1/JCRE	This SFR is covered by JCS.RunTimeExecution covering context switch and application selection.
FMT_MSA.1/JCVM	This SFR is covered by JCS.ByteCodeExecution requiring context switch for specific code execution and JCS.RunTimeExecution covering context switch and modification of the Currently Active Context according to given rules.
FMT_MSA.2/FIREWALL_JCVM	This SFR is addressed by JCS.RunTimeExecution covering object sharing.
FMT_MSA.3/FIREWALL	This SFR is addressed by JCS.RunTimeExecution covering object sharing.
FMT_MSA.3/JCVM	This SFR is addressed by JCS.RunTimeExecution covering object sharing.
FMT_SMF.1/JC	This SFR is addressed by JCS.RunTimeExecution covering context management and instance registration.
FMT_SMR.1/JC	This SFR is addressed by JCS.RunTimeExecution covering JCVM and JCRE roles.
FCS_CKM.1/ECDSA	This SFR is addressed by JCS.KeyManagement covering key generation.
FCS_CKM.1/GP-SCP	This SFR is covered by GP.SecureChannel.
FCS_CKM.4	This SFR is addressed by JCS.KeyManagement covering key deletion.
FCS_COP.1/TDES_MAC	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/AES_MAC	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services

	provided to applets through the Javacard API.
FCS_COP.1/ECDH	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/CRC	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/ECDSA_SIGN	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/ECKA_EG	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/GP-SCP	This SFR is covered by GP.SecureChannel.
FCS_COP.1/TDES_CIPHER	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/AES_CIPHER	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/RSA_SIGN	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/RSA_CIPHER	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/Hash	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FCS_COP.1/HMAC	This SFR is covered by JCS.CryptoAPI dealing with the cryptographic services provided to applets through the Javacard API.
FDP_RIP.1/ABORT	This SFR is addressed by JCS.EraseResidualData covering data erasure.
FDP_RIP.1/APDU	This SFR is addressed by JCS.EraseResidualData covering data erasure.
FDP_RIP.1/bArray	This SFR is addressed by JCS.OutOfLifeDataUndisclosure and JCS.EraseResidualData covering data erasure.
FDP_RIP.1/GlobalArray	This SFR is addressed by JCS.EraseResidualData covering data erasure.

FDP_RIP.1/KEYS	This SFR is addressed by JCS.EraseResidualData covering data erasure.
FDP_RIP.1/TRANSIENT	This SFR is covered by JCS.OutOfLifeDataUndisclosure managing the access control to transient object to be erased prior the erasure of the content in memory.
FDP_ROL.1/FIREWALL	This SFR is addressed by JCS.RunTimeExecution covering transaction rollback during specific operations.
FAU_ARP.1	This SFR is addressed by JCS.RunTimeExecution, JCS.Exception, JCS.Firewall, and OS.MemoryManagement covering exception handling with different specific operations.
FDP_SDI.2/DATA	This SFR is addressed by JCS.OwnerPIN, JCS.KeyManagement, OS.Atomicity and OS.MemoryManagement covering integrity handling with specific operations.
FPR_UNO.1	This SFR is addressed by JCS.OwnerPIN, JCS.KeyManagement, JCS.CryptoAPI and OS.MemoryManagement covering data handling with specific operations avoiding observation.
FPT_FLS.1/JC	This SFR is addressed by JCS.Exception, JCS.ByteCodeExecution, JCS.RunTimeExecution, and OS.Atomicity preserving a secure state when unexpected events occur during specific operations.
FPT_TDC.1	This SFR is covered by JCS.Package enforcing export check, CAP file translation and link specific operations.
FIA_ATD.1/AID	This SFR is covered by JCS.RunTimeExecution and GP.GPRegistry controlling applet registration and uninstallation.
FIA_UID.2/AID	This SFR is covered by GP.GPRegistry and JCS.RunTimeExecution managing user identity (package AID) during applet selection and identify associated context provided.
FIA_USB.1/AID	This SFR is covered by GP.GPRegistry and JCS.RunTimeExecution managing registration of each applet and associated package during its installation with its AID.
FMT_MTD.1/JCRE	This SFR is covered by JCS.RunTimeExecution offering services for applet registration and uninstallation managing associated access rights.
FMT_MTD.3/JCRE	This SFR is fully covered by JCS.RunTimeExecution managing presence and legacy of AID with ISO rules.
FDP_ACC.2/ADEL	This SFR is covered by GP.CardContentManagement, GP.GPRegistry and JCS.RunTimeExecution

	checking rules for applet instance uninstallation and deletion dependency rules.
FDP_ACF.1/ADEL	This SFR is covered by GP.CardContentManagement, GP.GPRegistry and JCS.RunTimeExecution checking rules for applet instance uninstallation and deletion dependency rules.
FDP_RIP.1/ADEL	This SFR is covered by GP.CardContentManagement and JCS.OutOfLifeDataUndisclosure by checking operations to avoid access to freed resources prior to its reuse.
FMT_MSA.1/ADEL	This SFR is covered by GP.GPRegistry, GP.CardContentManagement and JCS.RunTimeExecution responsible of checking rules concerning applet attributes, implicit and explicit selection rules prior to authorize deletion operation.
FMT_MSA.3/ADEL	This SFR is covered by JCS.RunTimeExecution and GP.CardContentManagement dealing with Security Attributes initialization, providing secure, restrictive default values for the security attributes of subject and objects involved in applet deletion.
FMT_SMF.1/ADEL	This SFR is covered by GP.CardContentManagement, GP.SecurityDomain and JCS.RunTimeExecution.
FMT_SMR.1/ADEL	This SFR is covered by GP.SecurityDomain maintaining the ISD and SDD roles responsible of applet deletion. This SFR is also covered by JCS.RunTimeExecution maintaining the JCRE role for applet uninstallation
FPT_FLS.1/ADEL	This SFR is covered by GP.GPRegistry, JCS.RunTimeExecution and OS.Atomicity preserving a secure state when unexpected events occur during package or instance deletion, managing the transaction part of the deletion operation by either rolling back, or completing it.
FDP_RIP.1/ODEL	This SFR is covered by JCS.EraseResidualData and OS.MemoryManagement ensuring that the content of deleted objects is erased upon the deletion and by JCS.OutOfLifeDataUndisclosure making unavailable for disclosure upon further reallocation of the freed space.
FPT_FLS.1/ODEL	This SFR is covered by JCS.RunTimeExecution and OS.MemoryManagement performing memory management to release no more used memory on unreferenced objects and

	preserves a secure state when unexpected events occur during object deletion.
FPT_FLS.1/GP	This SFR is addressed by JCS.Package, JCS.RunTimeExecution and GP.CardContentManagement covering the applet instance registration operations and associated error handling.
FDP_ROL.1/GP	This SFR is addressed by GP.CardContentManagement, GP.KeyLoading and OS.Atomicity.
FCO_NRO.2/GP	This SFR is covered by GP.SecureChannel managing the secure channel protocol where several checks are performed prior ELF or Key loading: * mutual authentication between the external entity (Issuer or Application provider) and the selected security Domain, including creation of a session key, * by the verification of a (chained) MAC that the Issuer or Application provider attaches to each file or data block sent, * by the erase of the session key at the end of the session.
FMT_SMR.1/GP	This SFR is covered by JCS.RunTimeExecution and GP.SecurityDomain managing the roles: S.OPEN, issuer, application provider, verification authority and controlling authority.
FMT_SMF.1/GP	This SFR is covered by GP.SecurityDomain and GP.SecureChannel.
FDP_ITC.2/GP-ELF	This SFR is covered by JCS.Package checking the binary compatibility of dependent packages using their version numbers and AIDs prior to installation operations.
FDP_ITC.2/GP-KL	This SFR is covered by GP.KeyLoading.
FPT_RCV.3/GP	This SFR is addressed by JCS.RunTimeExecution, OS.MemoryManagement, GP.GPRegistry and GP.CardContentManagement covering the applet instance erasure when applet instance registration operation fails.
FDP_IFC.2/GP-ELF	This SFR is covered by GP.CardContentManagement managing flow control for loading and installing application instances.
FDP_IFF.1/GP-ELF	This SFR is covered by GP.CardContentManagement managing flow control for loading and installing application instances.
FIA_UID.1/GP	This SFR is covered by JCS.RunTimeExecution and GP.SecurityDomain controlling accessible action prior identification and action when SD or application associated to SD are selected.

FIA_AFL.1/GP	This SFR is covered by GP.SecureChannel.
FIA_UAU.1/GP	This SFR is covered by JCS.RunTimeExecution and GP.SecurityDomain (as for FIA_UID.1/GP).
FIA_UAU.4/GP	This SFR is covered by GP.SecureChannel.
FDP_UIT.1/GP	This SFR is covered by GP.SecureChannel providing a session key generation. It ensures that the whole package or data has been correctly received.
FDP_UCT.1/GP	This SFR is covered by GP.SecureChannel, which provides confidentiality protection for sensitive data (such as secret keys).
FTP_ITC.1/GP	This SFR is addressed by GP.SecureChannel.
FPR_UNO.1/GP	This SFR is covered by JCS.RunTimeExecution and JCS.CryptoAPI.
FPT_TDC.1/GP	This SFR is addressed by GP.CardContentManagement, GP.SecureChannel and GP.KeyLoading.
FDP_IFC.2/GP-KL	This SFR is covered by GP.KeyLoading, GP.SecurityDomain and GP.SecureChannel.
FDP_IFF.1/GP-KL	This SFR is covered by GP.KeyLoading, GP.SecurityDomain and GP.SecureChannel.
FMT_MSA.1/GP	This SFR is covered by GP.SecureChannel providing an APDU flow control using the Command security level check according to Card Life cycle and type of APDU.
FMT_MSA.3/GP	This SFR is covered by GP.SecureChannel providing setting of the default value.
FDP_ACC.1/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FDP_ACF.1/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FMT_MSA.3/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FMT_SMR.1/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FMT_SMF.1/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FIA_ATD.1/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FTP_TRP.1/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FCS_COP.1/OS-UPDATE-DEC	This SFR is addressed by GP.OS-UPDATE.
FCS_COP.1/OS-UPDATE-VER	This SFR is addressed by GP.OS-UPDATE.
FPT_FLS.1/OS-UPDATE	This SFR is addressed by GP.OS-UPDATE.
FAU_SAS.1	This SFR is covered by OS.MemoryManagement
FPT_RCV.3/OS	This SFR is covered by OS.Atomicity.
FPT_RCV.4/OS	This SFR is covered by OS.MemoryManagement.

9 COMPOSITION WITH IC

9.1 Statement of compatibility – Threats part

IC Threats	Rationale
Part of [PP-84]	
BSI.T.Leak-Inherent	This threat is related to the information which is leaked from the TOE during usage of the Security IC in order to disclose sensitive data of the TOE. It is considered in the TOE evaluation.
BSI.T.Phys-Probing	This threat is related to physical probing of the TOE to disclose relevant information. It is considered in the TOE evaluation.
BSI.T.Malfunction	This threat is related to force malfunctions of the TSF due to environmental stress that could lower or bypass the implemented security mechanisms. It is considered in the TOE evaluation.
BSI.T.Phys-Manipulation	This threat is related to physical manipulation of the Security IC. It is covered by the IC evaluation.
BSI.T.Leak-Forced	This threat is related to information, which is leaked from the TOE during usage of the Security IC in order to disclose confidential user data of the composite TOE. It is covered by the IC evaluation.
BSI.T.Abuse-Func	This threat is related to the usage of functions of the TOE that are not allowed once the TOE Delivery and can affect the security of the TOE. It is considered in the TOE evaluation.
BSI.T.RND	This threat is related to the deficiency of random numbers. It is covered by the IC evaluation.
BSI.T.Masquerade_TOE	This threat is related to the IC masquerade. It is covered by the IC evaluation.
Added of [ST/IC]	
AUG4.T.Mem-Access	This threat is related to the memory access violation. The TOE implements memory access violation mechanisms based on the IC security policy. It is considered in the TOE evaluation.
JIL.T.Open_Samples_Diffusion	This threat is related to the diffusion of open samples. It is considered in the IC evaluation.
T.Confid-Applic-Code	It is considered in the TOE evaluation.
T.Confid-Applic-Data	It is considered in the TOE evaluation.
T.Integ-Applic-Code	It is considered in the TOE evaluation.
T.Integ-Applic-Data	It is considered in the TOE evaluation.

9.2 Statement of compatibility – OSPs part

IC OSPs	Rationale
Part of [PP-84]	
BSI.P.Process-TOE	This policy is related to the accurate unique identification during IC Development and Production. It is covered by the IC evaluation.
BSI.P.Lim_Block_Loader	Limiting and blocking the loader functionality for loading of TOE Software. It is covered by the ALC_DVS.2 activity of the TOE evaluation.
BSI.P.Ctlr_loader	Controlled usage to loader functionality. It is covered by the ALC_DVS.2 activity of the TOE evaluation.
Added of [ST/IC]	
AUG4.P.Add-Functions	It is considered in the TOE evaluation.

9.3 Statement of compatibility – Assumptions part

IC Assumptions	Rationale
Part of [PP-84]	
BSI.A.Process-Sec-IC	This assumption ensures the security of the delivery and storage of the IC. It is covered by the ALC_DVS.2 activity of the TOE evaluation.
BSI.A.Resp-Appl	This assumption ensures that security relevant data of the current TOE are properly treated according to the IC security needs. It is covered by the ADV_IMP.1 activity of the TOE evaluation.

9.4 Statement of compatibility – Security objectives for the environment part

IC OEs are separated in the following groups as defined in appendix 1.1 of [CC-COMP]:

- **IrOE**: IC OE being not relevant for the current TOE.
- **CfPOE**: IC OE being fulfilled by the current TOE automatically.
- **SgOE**: The remaining IC OE which shall be addressed by the current TOE.

IC OEs	Rationale
Part of [PP-84]	
BSI.OE.Resp-Appl	This objective deals with the treatment of TOE user data by the TOE itself. It is covered by the ADV_IMP.1 activity of the TOE evaluation. CfPOE
BSI.OE.Process-Sec-IC	This objective is covered by the IC evaluation and by the ALC_DVS.2 activity of the TOE evaluation. • During phases b, c: CfPOE • During phase e: SgOE
BSI.OE.Lim_Block_Loader	This objective is covered by the IC evaluation and by the ALC_DVS.2 activity of the TOE evaluation. • During phases b, c: CfPOE
BSI.OE.Loader_Usage	This objective is covered by the IC evaluation and by the ALC_DVS.2 activity of the TOE evaluation. • During phases b, c: CfPOE
BSI.OE.TOE_Auth	This objective is covered by the IC evaluation and by the ALC_DVS.2 activity of the TOE evaluation. During phases b, c: CfPOE
Added of [ST/IC]	
OE.Composite-TOE-Id	It is covered by the ALC_DVS.2 activity of the TOE evaluation. CfPOE
OE.TOE-Id	SgOE
OE.Enable-Disable-Secure-Diag	CfPOE
OE.Secure-Diag-Usage	CfPOE

9.5 Statement of compatibility – Security objectives part

IC Security objectives	Rationale
Part of [PP-84]	
BSI.O.Leak_inherent	This objective is covered by TOE evaluation.
BSI.O.Phys-Probing	This objective is covered by TOE evaluation.
BSI.O.Malfunction	This objective is covered by TOE evaluation.
BSI.O.Phys-Manipulation	This objective is covered by the IC evaluation.
BSI.O.Leak-Forced	This objective is covered by the IC evaluation.
BSI.O.Abuse-Func	This objective is covered by the TOE evaluation.
BSI.O.Identification	This objective is covered by the IC evaluation.
BSI.O.RND	This objective is covered by the IC evaluation.
BSI.O.Cap_Avail_Loader	This objective is covered by the TOE evaluation.
BSI.O.Authentication	This objective is covered by the TOE evaluation.
BSI.O.Ctrl_Auth_Loader	This objective is covered by the TOE evaluation.
Added of [ST/IC]	
JIL.O.Prot-TSF-Confidentiality	This objective is covered by the IC evaluation.
JIL.O.Secure-Load-ACode	This objective is covered by the TOE evaluation.
JIL.O.Secure-AC-Activation	This objective is covered by the TOE evaluation.
JIL.O.TOE-Identification	This objective is covered by the TOE evaluation.
O.Secure-Load-AMemImage	This objective is covered by the TOE evaluation.
O.MemImage-Identification	This objective is covered by the TOE evaluation.
AUG1.O.Add-Functions	This objective is covered by the IC evaluation.
AUG4.O.Mem-Access	This objective is covered by the TOE evaluation.
O.Firewall	This objective is covered by the TOE evaluation.

9.6 Statement of compatibility – SFRs part

IC SFRs are separated in the following groups as defined in appendix 1.1 of [CC-COMP]:

- **IP_SFR**: Irrelevant IC SFR not being used by the current TOE.
- **RP_SFR-SERV**: Relevant IC SFR being used by the current TOE to implement a security service with associated TSFI.
- **RP_SFR-MECH**: Relevant IC SFR being used by the current evaluation because of its security properties providing protection attacks to the TOE as a whole and are addressed in ADV_ARC. These required security properties are a result of the security mechanisms and services that are implemented in the IC.

IC SFRs	Rationale
Part of [PP-84]	
FRU_FLT.2	RP_SFR-MECH
FPT_FLS.1	RP_SFR-MECH
FMT_LIM.1	RP_SFR-MECH
FMT_LIM.2	RP_SFR-MECH
FAU_SAS.1	RP_SFR_SERV
FDP_SDC.1	RP_SFR-MECH
FDP_SDI.2	RP_SFR-MECH
FPT_PHP.3	RP_SFR-MECH
From "Leakage"	
FDP_ITT.1	RP_SFR-MECH
FPT_ITT.1	RP_SFR-MECH
FDP_IFC.1	RP_SFR-MECH
From "Weak cryptographic quality of Random Numbers"	
FCS_RNG.1	RP_SFR_SERV
From "Memory Access Violation and Correct operation"	
FDP_ACC.2/Memories	RP_SFR_SERV
FDP_ACF.1/Memories	RP_SFR_SERV
FMT_MSA.3/Memories	RP_SFR_SERV
FMT_MSA.1/Memories	RP_SFR_SERV
FMT_SMF.1/Memories	RP_SFR_SERV
From "Cipher scheme support"	
FCS_COP.1	RP_SFR_SERV
From "Abuse of loader functionality"	
FMT_LIM.1/Loader	IP_SFR
FMT_LIM.2/Loader	RP_SFR_SERV
FTP_ITC.1/Loader	RP_SFR_SERV
FDP_UCT.1/Loader	RP_SFR_SERV
FDP_UIT.1/Loader	RP_SFR_SERV
FDP_ACC.1/Loader	RP_SFR_SERV
FDP_ACF.1/Loader	RP_SFR_SERV
From "Masquerade"	
FIA_API.1	RP_SFR_SERV
From "Correct Loader operation"	
FMT_MSA.3/Loader	RP_SFR_SERV
FMT_MSA.1/Loader	RP_SFR_SERV
FMT_SMR.1/Loader	RP_SFR_SERV
FIA_UID.1/Loader	RP_SFR_SERV
FIA_UAU.1/Loader	RP_SFR_SERV
FMT_SMF.1/Loader	RP_SFR_SERV
FPT_FLS.1/Loader	RP_SFR_SERV
From "Lack of TOE identification"	
FAU_SAR.1/Loader	RP_SFR_SERV
FAU_SAS.1/Loader	RP_SFR_SERV
From "Abuse of Secure Diagnostic functionality"	
FTP_ITC.1/Sdiag	RP_SFR_SERV
FAU_SAR.1/Sdiag	RP_SFR_SERV
FMT_LIM.1/Sdiag	RP_SFR_SERV
FMT_LIM.2/Sdiag	RP_SFR_SERV

10 REFERENCES, GLOSSARY AND ABBREVIATIONS

10.1 External references

Reference	Title
[ISO7816]	Identification cards – Integrated circuit(s) cards with contacts - Books 1 to 9
[CC-1]	Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model, CCMB-2017-04-001, version 3.1 revision 5, April 2017.
[CC-2]	Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements, CCMB-2017-04-002, version 3.1 revision 5, April 2017.
[CC-3]	Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements, CCMB-2017-04-003, version 3.1 revision 5, April 2017.
[CC-COMP]	Common Criteria Supporting Document, Mandatory Technical Document – Composite product evaluation for Smart Cards and similar devices, version 1.5.1, May 2018.
[CIC]	Common Implementation Configuration v2.0 (GPC_GUI_080)
[EUPP]	TCA eUICC Profile Package Interoperable Format Test Specification v2.3.1, September 2020
[11]	[GPCS] Global Platform Card Specification v2.3.1 (GPC_SPE_034), March 2018 – ref [11] in [PP/0100] and amendments • [Amd A] Amendment A - Confidential Card Content Management, v1.2 (GPC_SPE_007) • [Amd B] Amendment B - Remote Application Management over HTTP, v1.2 (GPC_SPE_011) – ref [13] in [PP/0100] • [Amd D] Amendment D - Secure Channel Protocol 03, v1.2 (GPC_SPE_014) • [Amd E] Amendment E - Security Upgrade for Card Content Management for ECDSA/ECC, v1.1 • [Amd F] Amendment F - Secure Channel Protocol '11' (SCP11c), v1.3
[12]	SCP80 ETSI TS 102 225, ETSI TS 102 226 – ref [12] in [PP/0100]
[JC] [JCAPI3] [JCVM3] [JCRE3]	Java Card Specification v3.1.0, November 2019 Java Card 3 Platform - Java Card API, Classic Edition, Version 3.1.0, November 2019 Java Card 3 Platform - Virtual Machine Specification, Classic Edition, Version 3.1.0, November 2019 Java Card 3 Platform - Runtime Environment Specification, Classic Edition, Version 3.1.0, November 2019
[JCBV]	Java Card 3.1.0 Off-card Verifier and onwards
[PP-84]	Security IC Platform Protection Profile with Augmentation Packages version 1.0, February 2014, BSI-CC-PP-0084-2014
[PP-eUICC]	Embedded UICC for Consumer Devices Protection Profile version 1.0, June 2018, BSI-CC-PP-0100-2018 (SGP.25 v1.0 by GSMA)
[PP-JCS]	Java Card System – Open Configuration Protection Profile version 3.1, April 2020, BSI-CC-PP-0099-V2-2020 – ref [01] in [PP/0100]
[PP-GP]	Secure Element Protection Profile version 1.0, February 2021, GPC_SPE_174
[SGP.02]	Remote Provisioning Architecture for Embedded UICC Technical Specification – ref [03] in [PP/0100]
[SGP.06]	eUICC Security Assurance Principles, version 2.1, March 2024
[SGP.07]	eUICC Security Assurance Methodology, version 2.1, March 2024
[SGP.21]	Architecture Specification, version 2.5, November 2022
[SGP.22]	RSP Technical Specification, version 2.5, May 2023 – ref [24] in [PP/0100]
[SGP.23]	RSP Test Specification, version 1.14, July 2023
[SGP.24]	SGP.24 Compliance Process, Version 2.5, March 2023

Reference	Title
[ST/IC]	Security Target for Composition ST54L A01: • SMD_ST54L_ST_22_001 Rev A02.1 February 2024
[GUIDES/IC]	List of documents applicable to the certified IC: • NFC controller and secure element system in package DS_ST54L V3 • ST54L_SE OS developer's guide, User Manual UM_ST54L_SE V4 • ST54L_SE Firmware V3 –User Manual, UM_ST54L_SE_FWv3 V3 • Arm Cortex-M35P Processor Technical Reference Manual 100883_0102_00_en r1p2 • Security guidance of the ST54L _SE secure MCU subsystem, Application note AN_SECU_ST54L_SE V1.0 • Random number Generator V1.4, User Manual UM_ST54L_SE_TRNG14 V5
[CERT/IC]	NSCIB-CC-2300182-01
[VER]	Global Platform Card Composition Model, Security Guidelines for Basic Applications (GPC_GUI_050, v2.0)
[IAD]	D1616769_v9.2 - PLATFORM_IAD For Connected eSE 5.3.4 v1.2 Rev 02 - Impact Analysis Document

10.2 Internal references

Reference	Title
[GUIDES_Rev02]	List of documents applicable to the certified **Connected eSE 5.3.4 v1.2 Rev 02** configuration: • Connected_eSE 5.3.4 - APDU Guide (ref D1596336B) – 8 March 2024 • Connected_eSE 5.3.4 - Applet Dev Guide (ref D1542793A) – April 03, 2023 • Connected eSE 5.3.4_Pdt User's Guide_ (ref D1596339B) - November 2024 • D1620263 Version 1.0 - Operational guidance on CC platforms for Verification Authority – Connected eSE 5.3.4 v1.2 • D1258682 C03 - Application Verification for Certified Secure Elements - External Procedure • D1344508 A04 - Patch Loading Management for Certified Secure Elements - External Procedure • D1516176 v3.4 - Guidance for Secure application development on Thales Embedded Secure Solutions • D1546158 v1.1 - Security Technical Note for the guides • D1624075 v1.2 eSA-Operational guidance of Thales Connected eSE 5.3.4_V1.2 • D1624077 v1.1 eSA-Preparative guidance of Thales Connected eSE 5.3.4_V1.2 • D1639332 v1.3 - PLATFORM Identification Configurability For Connected eSE 5.3.4 v1.2 Rev 02

10.3 Glossary

Term	Definition
Application	Instance of an Executable Module after it has been installed and made selectable
Controlling Authority	A Controlling Authority is entity independent from the OEM represented on the eUICC and responsible for securing the keys creation and personalization of the Supplementary Security Domains.
DAP Block	Part of the Load File used for ensuring Load File Data Block verification
DAP Verification	A mechanism used by a Security Domain to verify that a Load File Data Block is authentic
Issuer Security Domain	The primary on-card entity providing support for the control, security, and communication requirements of the card administrator
Profile	Security Domains, UICC file system and secure objects (Keys, PIN codes…) formatted as defined by [EUPP]. A Profile can be downloaded from RSP Servers onto a eUICC by end user consent, as defined by [SGP.21] [SGP.22].

Term	Definition
RSP Servers	GSMA-defined SM-DP+ and SM-DS servers. Used to distribute a Profile to the end user.
Security Domain	On-card entity providing support for the control, security, and communication requirements of an off-card entity (e.g. the Profile Issuer, an Application Provider or a Controlling Authority)
Supplementary Security Domain	A Security Domain other than the Issuer Security Domain dedicated to Application provider.
Verification Authority	The Verification Authority (VA), is a trusted third party represented on the (U)SIM card, acting on behalf of the OEM and responsible for the verification of application signatures (mandated DAP) during the loading process.

10.4 Abbreviations

CC	Common Criteria
HW	Hardware
ISD	Issuer Security Domain
ISD-P	Issuer Security Domain Profile (see [SGP.22])
ISD-R	Issuer Security Domain Root (see [SGP.22])
LPAd	Local Profile Assistant on Device (see [SGP.22])
OEM	Original Equipment Manufacturer
OTA	Over-The-Air
PP	Protection Profile
REE	Rich Execution Environment (e.g. Android, iOS, Linux, Windows, etc.)
RMA	Return Merchandise Authorization (i.e. return a product under warranty for a replacement, refund, repair)
ST	Security Target
SW	Software
TOE	Target of Evaluation
VA	Verification Authority
CC	Common Criteria

END OF DOCUMENT