Agence Nationale de la Sécurité des Systèmes d'Information

Protection Profile

Embedded Software for Smart Secure Devices

Basic and Extended Configurations

Version 1.0

November 27th, 2009

Table of contents

Table of figures

Table of tables

1 Protection Profile Introduction

1.1 Protection Profile Identification

This document holds two Protection Profiles, identified by ANSSI-CC-PP-ESforSSD_Basic and ANSSI-CC-PP-ESforSSD_Extended, as indicated below.

Title:	Protection Profile, Embedded Software for Smart Secure Devices – Basic Configuration
Identification	ANSSI-CC-PP-ESforSSD_Basic
Editor:	Trusted Labs
Date:	November 27th, 2009
Version:	1.0
Sponsor:	ANSSI
CC Version:	3.1 Revision 3

Title:	Protection Profile, Embedded Software for Smart Secure Devices – Extended Configuration
Identification	ANSSI-CC-PP-ESforSSD_Extended
Editor:	Trusted Labs
Date:	November 27th, 2009
Version:	1.0
Sponsor:	ANSSI
CC Version:	3.1 Revision 3

In the following, "this Protection Profile" stands for the Protection Profile collection composed of ANSSI-CC-PP-ESforSSD_Basic and ANSSI-CC-PP-ESforSSD_Extended.

1.2 Protection Profile Presentation

This Protection Profile replaces the Protection Profile "Smart Card Integrated Circuit with Embedded Software" [PP9911] certified by French Scheme ANSSI in 1999. It meets current needs and technological trends in smart secure devices, e.g. smartcards. Joint work with actors from the smartcard industry and with specialized evaluation laboratories has been carried out in order to have the most accurate inputs and actual constraints. The members of the Consortium that supported the edition work are: Gemalto, Oberthur Technologies, Sagem Orga, BMS, GIE Cartes Bancaires, CEA LETI and SERMA Technologies.

This Protection Profile applies to smart secure devices (hereafter the "product" or the "device") composed of a Security Integrated Circuit (hereafter the "Security IC" or simply the "IC") and of Embedded Software (hereafter the "native OS" or simply the "OS") running on top of this IC. The Security IC provides processing units, security components, random number generator, I/O ports, volatile and non-volatile memories. The Embedded Software implements Operating System functionalities such as secure boot, memory management, life cycle management and, potentially, applicative behaviour.

The products targeted by this Protection Profile are of two kinds:

- "Integrated products" where the Embedded Software consists of native code that implements both OS and applicative behaviour without demarcation between them;

- "Layered products" where the Embedded Software consists of an "OS Layer", potentially with integrated applicative behaviour, and an "Application Layer" on top of it. The OS provides a separation mechanism between itself and the Application Layer as well as services to the Application Layer.

This Protection Profile focuses on the security requirements for the OS, which constitutes the TOE; the Security IC is considered as the environment of the OS, covered by security objectives. Nevertheless, any smart secure device evaluation against this PP shall comprehend the whole product including both the Security IC and the OS: the security target of the product shall enforce the security objectives for the IC stated in this Protection Profile by means of security requirements for the IC.

The evaluation of the product may be "composite" according with security assurance requirements in [COMP][1], provided the IC has been evaluated separately. This Protection Profile does not require any formal compliance to a specific hardware PP for the IC evaluation but those ICs evaluated against [PP0035] fully meet the objectives. However, composition with an already certified IC is not mandatory. In the case where the IC has not been certified, the product evaluation shall address both the IC and the OS at the same time. Requirements on vulnerability assessment stated in [CC AP] document apply in both cases.

This Protection Profile defines two TOE configurations, Basic and Extended, that map to the kinds of products identified above:

- Basic TOE: There is no separation between the OS and the applications. The Basic TOE implements security features for its own purposes (cf. section 2.4.1). This configuration corresponds to integrated products;
- Extended TOE: The OS implements a separation mechanism between itself and the Application Layer. The Extended TOE implements security features for its own purposes and potentially for the Application Layer (cf. section 2.4.2). This configuration corresponds to layered products.

Each TOE configuration gives rise to a Protection Profile configuration, with unique identification:

- "ANSSI-CC-PP-ESforSSD_Basic Basic Configuration" addresses integrated products with Basic TOE where the OS and the applicative behaviour are not separated;
- "ANSSI-CC-PP-ESforSSD_Extended Extended Configuration" addresses layered products with Extended TOE where the OS provides a separation mechanism between itself and the Application Layer that runs on top of it.

This Protection Profile requires "demonstrable" conformance.

Security Targets or Protection Profiles conformant to this Protection Profile can enlarge the perimeter of the chosen TOE configuration with additional functionalities like, for instance, authentication mechanisms, post-delivery loading of code and data, secure communication protocols[2].

The evaluation of this Protection Profile has been performed by the French ITSEF CEA-LETI. The PP has been certified by French Scheme ANSSI.

The organisation of the document is as follows: sections, figures and tables that are specific to ANSSI-CC-PP-ESforSSD_Basic, respectively ANSSI-CC-PP-ESforSSD_Extended, are tagged with the identifier ANSSI-CC-PP-ESforSSD_Basic, respectively ANSSI-CC-PP-ESforSSD_Extended; sections, figures and tables that apply to both protection profiles do not hold any tag.

[1] The Security Assurance Requirements in [COMP] come in addition to the EAL specified in this Protection Profile, especially the requirements related to the recommendations from the Security IC evaluation.

[2] This Protection Profile has been designed to ease composite evaluations with applicative Protection Profiles like "Electronic Purse" Protection Profile [PPEP] and "Java Card System" Protection Profiles [PPJCS].

1.3 References

[CC1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model. Version 3.1, revision 3, July 2009. CCMB-2009-07-001.

[CC2] Common Criteria for Information Technology Security Evaluation, Part 2: Security functional requirements. Version 3.1, revision 3, July 2009. CCMB-2009-07-002.

[CC3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements. Version 3.1, revision 3, July 2009. CCMB-2009-07-003.

[CEM] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology. Version 3.1, revision 3, July 2009. CCMB-2009-07-004.

[COMP] Composite product evaluation for smartcards and similar devices Version 1.0, September 2007, CCDB-2007-09-01.

[CRYPTO] Mécanismes cryptographiques. Règles et recommendations concernant le choix et le dimensionnement des mécanismes cryptographiques. Version 1.11, 24 Octobre 2008, ANSSI. (This version or later applicable one.)
Gestion des clés cryptographiques. Règles et recommendations concernant la gestion des clés utilisées dans des mécanismes cryptographiques. Version 1.10, 24 Octobre 2008. ANSSI. (This version or later applicable one.)

[CC AP] Common Criteria Supporting Document - Mandatory Technical Document - Application of attack potential to smart-cards, reference CCDB-2009-03-001 version 2.7 revision 1, March 2009, or current applicable version.

[PP0035] Security IC Platform Protection Profile, Version 1.0, June 2007

[PP9911] Smart Card Integrated Circuit with Embedded Software Protection Profile, Version 2.0, June 1999

[PPEP] Protection Profile – Electronic Purse, BMS – SFPMEI.

[PPJCS] Java Card System Protection Profiles, Closed Configuration and Open Configuration.

[PPSAM] Protection Profile – Secure Access Module for Electronic Money System, BMS – SFPMEI.

2 TOE overview

This chapter defines the type of the Target of Evaluation (TOE), presents the TOE in typical product architectures and describes its main security features and intended usages.

2.1 TOE Type

The TOE type is the native software layer or Operating System embedded in an Integrated Circuit designed for a Secure Smart Device. The TOE may integrate applicative behaviour and does not comprise the IC. This TOE defines the perimeter of the security requirements stated in this Protection Profile but does not define the perimeter of an actual evaluated product that must include the Security IC.

The Basic TOE comprises:

1. OS and integrated applicative native code and data stored in the Security IC memories;
2. TOE guidance delivered to the User of the product.

The Extended TOE comprises:

1. OS native code and data stored in the Security IC memories;
2. optional integrated applicative native code and data stored in the Security IC memories;
3. TOE guidance delivered to the Application Developer;
4. TOE guidance delivered to the User of the product if the TOE integrates applicative behaviour (item 2 above).

Unless stated otherwise, TOE stands for both TOE configurations, without guidance.

2.2 Basic TOE in the integrated product architecture

The diagram in Figure 1 shows the Basic TOE (native OS with applicative behaviour) in the framework of a Smart Secure Device composed of two layers:

- the IC provides low-level security mechanisms and resources to the Basic TOE, including CPU, memories, communication interfaces, Random Number Generator, sensors;
- the TOE manages the IC resources, monitors IC detectors, implements cryptographic operations, etc. (cf. section 2.4.1 for a detailed description of Basic TOE security features).

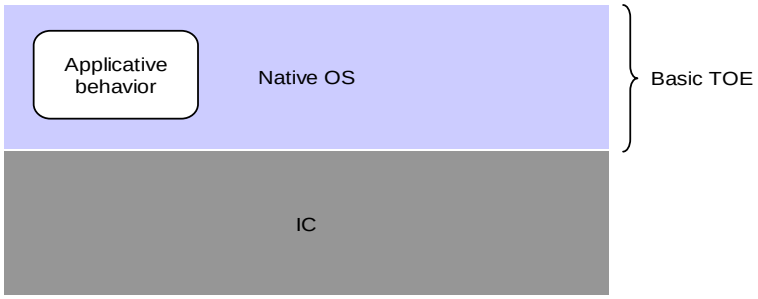

Figure 1: Integrated Product with Basic TOE – ANSSI-CC-PP-ESforSSD_Basic

2.3 Extended TOE in the layered product architecture

The diagram in Figure 2 shows the Extended TOE (native OS with separation mechanism and, potentially, with applicative behaviour) in the framework of a Smart Secure Device composed of three layers:

- the Security IC layer provides low-level security mechanisms and resources to upper layers;
- the Native OS layer manages the IC resources, monitors IC security detectors, implements a separation mechanism between itself and the Application Layer, provides security services to the Application Layer, and provides functionality to the product end users through the applicative behaviour, if any (cf. section 2.4.2 for a detailed description of Extended TOE security features);
- the Application Layer uses the Extended TOE to access resources and services and implements specific functionalities provided to the end users.

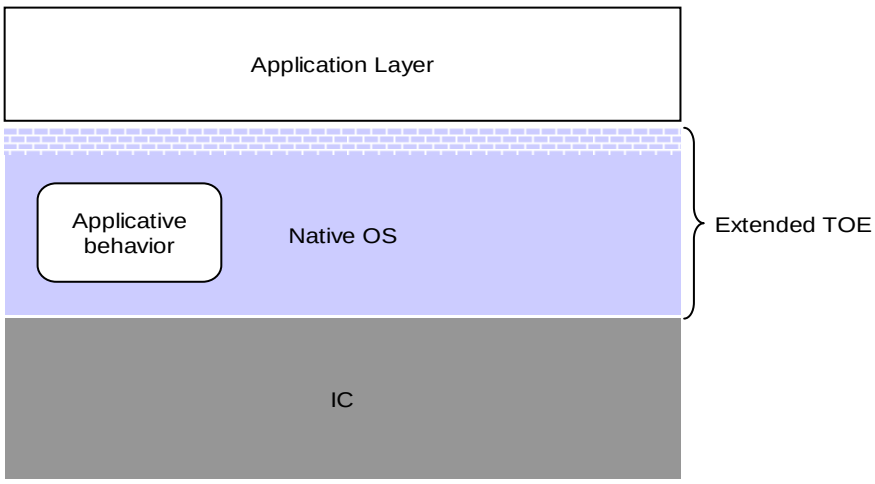

Figure 2: Layered Product with Extended TOE – ANSSI-CC-PP-ESforSSD_Extended

The Protection Profile for the Extended Configuration does not state any requirement on the Application Layer. From the point of view of the TOE, the Application Layer looks like a single application. The loading and personalisation of the Application Layer may occur at the same phases as the loading and personalization of the Extended TOE (cf. section 2.7).

The Extended TOE supports the implementation of native devices, where the Application Layer contains only native code, or interpreted devices, for instance a Java Card where the Application Layer consists of a JC System (JC Virtual Machine, JC Runtime Environment and JC APIs) and of JC applets, as shown in Figure 3, or even mixed devices where the Application Layer holds a Java Card System and other native applications. Nevertheless, the Protection Profile for the Extended Configuration does not address Java Card specific security features; hence the evaluation of Java Card enabled products shall mostly rely on Java Card System Protection Profiles [PPJCS].

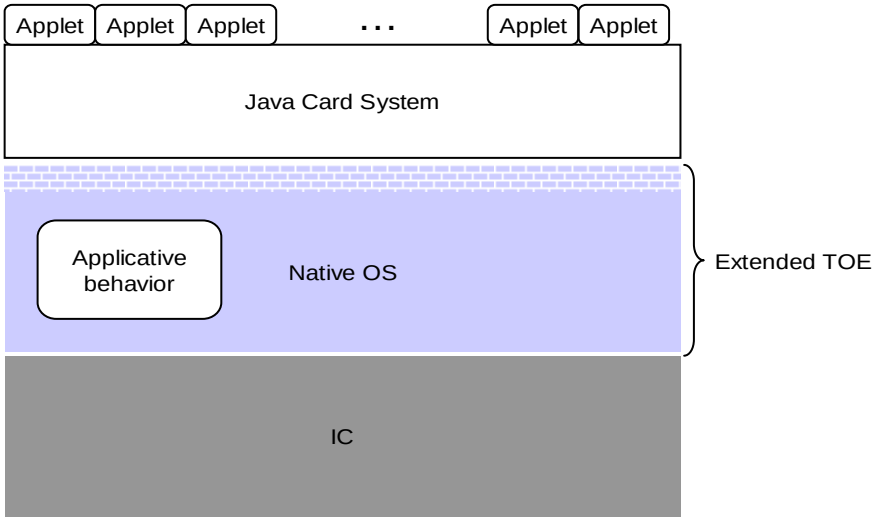

Figure 3: Java Card on a Layered Product – ANSSI-CC-PP-ESforSSD_Extended

2.4 TOE Security Features

The main goal of the TOE is to provide a secure execution environment to native applications with critical assets that need to be protected against unauthorised disclosure and/or modification. The TOE shall avoid bypassing, abuse or tampering of its services that could compromise the security of the assets.

Section 2.4.1 presents the security features of the Basic TOE configuration in the Final Usage phase (cf. section 2.7). Section 2.4.2 presents the security features of the Extended TOE configuration.

Application note: Security Targets that are conformant to this PP shall adapt these descriptions to the characteristics of the product under evaluation.

2.4.1 Security features of the Basic TOE configuration

The Basic TOE shall provide secure implementation of the following features:

Boot at power-up

The TOE shall boot after the Security IC has successfully powered-up. The TOE boot operations shall ensure the correct initialization of the TOE functionalities.

Application Note: The Security Target author shall describe the boot operations performed by the OS in complement to the Security IC operations.

Monitoring

The TOE shall monitor all the events generated by the Security IC physical detectors that are made available to the TOE (e.g. out-of-range voltage, temperature, frequency, active shield, memory aging) and it shall provide automatic answers that put the product in a secure state upon potential security violations.

Application Note: The Security Target author shall describe the behaviour of the TOE upon detection of a potential security violation by the Security IC.

Execution environment

The TOE shall provide a secure execution environment based on the secure operation of CPU (e.g. well known instructions, no hidden or unspecified code) that controls the execution flow, detects and reacts to potential security violations. Moreover, the TOE shall ensure that patches installed before delivering, if any, cannot be bypassed.

The TOE identification shall take into account the patches installed before delivery.

Application note:
- *the patches shall be uniquely identified as components of the TOE within the framework of ALC_CMS.4 evaluation task;*
- *the Security Target author shall describe the principles of the security execution and the behaviour of the TOE upon potential security violations.*

Memory management

The TOE shall manage the persistent and volatile memories of the product according to the capacities of the underlying Security IC so as to control access to sensitive content protected by the TOE. TOE memory management services may include memory allocation/deallocation, direct or handle-based access control, integrity checks, enciphering (on-the-fly), pagination, etc.

The TOE shall ensure that no residual information is available from memories, for instance, cleaning persistent memory upon allocation and deallocation and wiping of volatile memory upon sensitive operations.

Application Note: The Security Target author shall describe the memory management features of their product in complement of the underlying Security *IC mechanisms.*

I/O management

The TOE shall manage Input/Output interfaces together with the Security IC. Any kind of external interface is allowed (contact, contactless, USB, etc.). The TOE shall provide the same security level whatever the communication mode is.

The TOE shall capture all communication requests and shall be able to decide whether to accept or deny them.

The TOE shall prevent leakage of data through the I/O buffers by ensuring that only genuine request can output data via the I/O interfaces.

The TOE shall control I/O buffers in order to prevent data leakage.

Application Note:
- *the evaluator's vulnerability analysis performed in the framework of the evaluation task AVA_VAN.5 shall demonstrate that the communication mode(s) handled by the product do not impact the security of the product;*
- *the Security Target author shall describe the I/O management by the OS in complement to the* Security *IC.*

Life cycle management

The TOE shall manage its life cycle and shall provide a secure transition mechanism between states; in particular, the mechanism shall prevent re-entering irreversible states. The TOE shall prevent abuse of functionalities that are available only at certain states in the TOE life cycle.

Application Note: The Security Target author shall describe all the states of the TOE life cycle as well as all the transitions allowed between states and she/he shall indicate the set of functionalities available in each state.

Cryptographic operations

The TOE shall provide a secure implementation of all the cryptographic operations used by the OS itself (e.g. for memory encryption) and/or by the integrated application (e.g. for signature or communication protocols). These operations may correspond to cryptographic primitives (e.g. DES, SHA) or to complex functions (e.g. signature generation).

Application Note: The Security Target author shall provide

- *the list of cryptographic operations that play a role in the security of the product together with the references to the corresponding standards and the underlying hardware cryptographic co-processors, if applicable;*
- *the list of individual keys managed by the TOE and the operations that apply to each of them.*

Key management

The TOE shall provide secure generation, destruction, replacement and storage of cryptographic keys according to the specification of the product.

Application Note: The Security Target author shall provide the list of key management operations of the product (it is not mandatory to implement all generation, destruction and replacement) together with the references to the methods and standards used, and the keys they apply to.

Random numbers

The TOE shall provide random numbers. The Random Number Generation shall be conformant to the quality requirements of the national schemes (such as [CRYPTO] in France).

Application Note: The Security Target author shall describe the characteristics of the random numbers provided by the TOE, resulting from an appropriate combination of hardware and software mechanisms.

Atomic operations

The TOE shall provide means of performing atomic operations, for instance, writing or erasing of individual or multiple memory locations. The TOE shall guarantee that atomic operations are either performed completely or have no effect in case of interruption.

Application Note: The Security Target author shall identify the atomic operations and describe the principles underlying the atomicity mechanism.

2.4.2 Security features of the Extended TOE configuration

The Extended TOE configuration shares all the security features of the Basic TOE configuration:

- secure boot at power-up;
- security monitoring;
- secure execution;
- memory management;
- I/O management
- life cycle management;
- cryptographic operations;
- key management;
- random numbers;
- atomic operations.

The Extended TOE configuration has two additional security features:

Separation mechanism

The TOE shall implement a separation mechanism between itself and the Application Layer that runs on top of it, e.g. separate execution domains and memory contexts for the TOE and the Application Layer.

Application Note: The Security Target author shall describe the principles of the separation mechanism implemented by the TOE, which may rely on IC mechanisms.

Services to the Application Layer

The TOE provides security services to the Application Layer through the TOE API, which may consist, for instance, of:

- API for cryptographic operations;
- API for key management;
- API for atomic transaction;
- API for RNG.

These security services rely on security features of the Basic TOE configuration. For instance, if the Extended TOE provides a key management API to the Application Layer, this means that the key management feature shall manage Application Layer keys also.

Application Note: The Security Target author shall list the actual TOE APIs of the product available to the Application Layer, together with the references to the corresponding standards if applicable.

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

The only non-TOE component available to the TOE is the Integrated Circuit of the product. A typical Security IC consists of:

- processing units (CPU, accelerators);
- I/O ports (contact, contactless, USB, etc.);
- volatile and non-volatile memories;
- Random Number Generator;
- detectors of operational conditions;
- dedicated software.

The Security IC shall

- provide Random Number Generator;

- resist to attackers with high-attack potential according to [CC AP] characterisation, in particular, the IC shall resist to:
    - leakage attacks;
    - intrusive (e.g. probing, fault injection) and non-intrusive (e.g. SPA, DPA, EMA) attacks;
    - operational conditions manipulation (voltage, clock, temperature, etc.);
    - physical attacks aiming at modification of the Security IC content or behaviour.

Typically, an IC evaluated against [PP0035] fulfils these requirements.

2.6 TOE Usage

The TOE is intended to meet requirements of applications with sensitive assets, in particular:
- banking and finance credit / debit smartcards, electronic purse (stored value cards) and electronic commerce, loyalties;
- network based transaction processing such as mobile phones (GSM SIM cards), pay-TV (subscriber and pay-per-view cards), communication highways (Internet access and transaction processing);
- transport and ticketing market (access control cards);
- governmental cards (electronic passports, ID-cards, health-cards, driver license, etc.);
- multimedia commerce and Intellectual Property Rights protection.

2.7 TOE Life Cycle

The TOE life cycle is part of the global product life cycle that goes from product development to its usage by the final user.

The product life cycle phases are those detailed in Figure 4. We refer to [PP0035] for a thorough description of Phases 1 to 7:
- Phases 1 and 2 compose the product development: Embedded Software (IC Dedicated Software, OS and potentially Application Layer) and IC development;
- Phases 3 and 4 correspond to IC manufacturing and packaging, respectively. Some IC pre-personalisation steps may occur in Phase 3;
- Phase 5 concerns the embedding of software components within the IC;
- Phase 6 is dedicated to the product personalisation prior final use;
- Phase 7 is the product operational phase.

The TOE life cycle is composed of four stages:
- TOE development;
- TOE storage, pre-personalisation and testing;
- TOE personalisation and testing;
- TOE final usage.

TOE storage is not necessarily a single step in the TOE life cycle since the TOE can be stored in parts. TOE delivery occurs before TOE storage and may take place more than once if the TOE is delivered in parts. The patches are special TOE parts. There is no patch or application loading in the field in any of the PP configurations.

The TOE stages relate to the typical smartcard life cycle phases as shown in Figure 4 below.

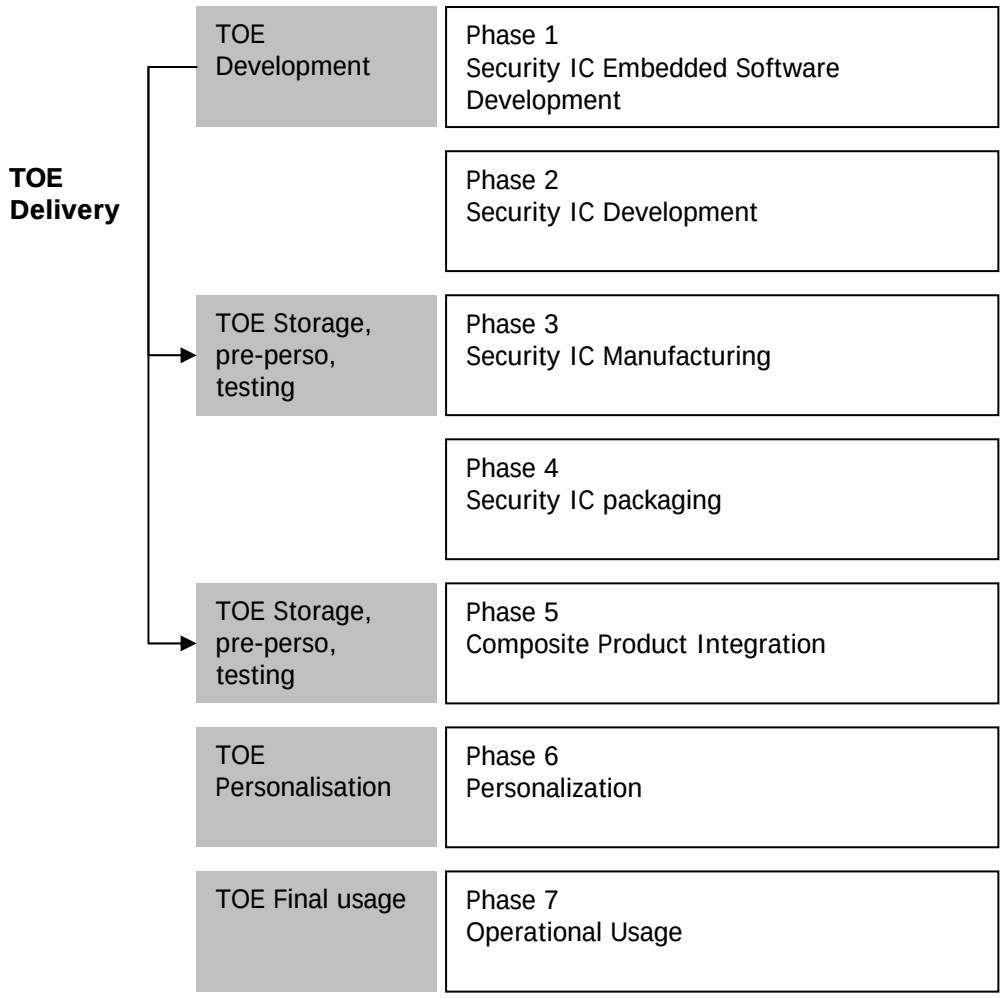

Figure 4: TOE Life Cycle within Product Life Cycle

TOE Development is performed during Phase 1. This includes TOE conception, design, implementation, testing and documentation. The TOE development shall fulfil requirements of the final product and recommendations of the Security IC user guidance. The TOE development shall occur in a controlled environment that avoids disclosure of source code, data and any critical documentation and guarantees the integrity of these elements. The evaluation of a product against this PP shall include the TOE development environment.

The delivery of the TOE in binary code format may occur either during Security IC Manufacturing (Phase 3) or during Composite Product Integration (Phase 5). It is also possible that part of the TOE is delivered in Phase 3 and the rest is delivered in Phase 5. Delivery and acceptance procedures shall guarantee the authenticity, the integrity and the confidentiality of the exchanged pieces according to their sensitivity. TOE delivery usually involves encrypted signed sending and supposes the previous exchange of public keys. The evaluation of a product against this PP shall include the delivery process.

In Phase 3, the Security IC Manufacturer may store, pre-personalize the TOE and potentially conduct tests on behalf of the TOE developer. The Security IC Manufacturing environment shall protect the integrity and confidentiality of the TOE and of any additional TOE material (e.g. test suites) according to their sensitivity. The evaluation of a product against this PP shall include the whole Security IC manufacturing environment, in particular those locations where the TOE is accessible for installation

or testing. If the Security IC has already been certified (e.g. against [PP0035]) there is no need to perform the evaluation of the whole Security IC manufacturing again.

In Phase 5, the Composite Product Integrator may store, pre-personalize the TOE and potentially conduct tests on behalf of the TOE developer. The Composite Product Integration environment shall protect the integrity and confidentiality of the TOE and of any additional TOE material (e.g. test suites) according to their sensitivity. Note that (part of) TOE storage in Phase 5 implies the delivery of the product after Phase 5. Hence, the evaluation of such product against this PP shall also include the Composite Product Integrator environment(s).

The TOE is personalized in Phase 6. If the product delivery point is at the end of Phase 6, then the personalization environment shall be included in a product evaluation. This means that some of the product personalization operations may require a controlled environment (secure locations, secure procedures and trusted personnel). The product shall be tested and all sensitive material including personalization data, test suites and documentation shall be protected from disclosure and modification.

The TOE final usage environment coincides with the final usage environment of the product. The TOE and the product shall provide the full set of security functionalities to avoid abuse of the product in this environment.

In the framework of a layered product with an Extended TOE, the loading and personalisation of the Application Layer may occur at the same phases as the loading and personalization of the Extended TOE.

Application note: The Security Target author shall specify the life cycle of the product, the TOE delivery point and the product delivery point. The product delivery point may arise at the end of Phase 3, 4, 5 or 6. Note that TOE delivery precedes product delivery. During product evaluation against this Protection Profile, the ALC security assurance requirements apply to the whole product life cycle up to delivery; the evaluation shall concern all the Security *IC and TOE environments up to the product delivery.*

3 Conformance claims

3.1 Conformance claim to CC

This Protection Profile is CC Part 2 conformant [CC2] and CC Part 3 [CC3] conformant.

3.2 Conformance claim to a package

The minimum assurance level for the evaluation of a Smart Secure Device comprising a TOE conformant to this PP is EAL4 augmented with AVA_VAN.5 and ALC_DVS.2 components.

3.3 Conformance claim of the PP

This PP does not claim conformance to any another PP.

3.4 Conformance claim to the PP

The conformance to this PP, required for the Security Targets and Protection Profiles claiming conformance to it, is **demonstrable**, as defined in CC Part 1 [CC1].

4 Underlying security model

This Protection Profile addresses products composed of a Security IC and an OS – potentially with applicative behaviour - but deals with these two entities differently:

1. the Security Problem Definition (SPD) concerns the whole product; the threats cover attacks against the product in its final usage phase and the assumptions and OSP address security aspects from product delivery up to the final usage phase;
2. the objectives for the TOE and the SFRs concern the OS - and its applicative behaviour if any- only, according with the definition of the TOE given in section 2.1;
3. the security objectives for the TOE operational environment concern the Security IC and the product operational environment. The properties of the Security IC are stated as security objectives for the environment, without SFR counterpart. This allows to state generic security goals for the Security IC that shall be mapped to specific requirements in actual product evaluations;
4. the SAR of this PP does not only apply to the TOE but to the whole product under evaluation. This is due to the fact that pure TOE – only software without considering the hardware on which it is embedded- evaluation is not allowed; the TOE shall always be evaluated with a specific IC.

Table 1 summarises the scope of each CC concept; the number in brackets refers to the items above.

CC concept	**OS (TOE)**	**IC**	**Product (IC + OS) Operational Environment**
Assets	x (1)	x (1)	
Threats	x (1)	x (1)	
Assumptions			x (1)
OSP			x (1)
Objectives for the TOE	x (2)		
Objectives for the TOE Environment		x (3)	x (3)
SFR	x (2)		
SAR	x (4)	x (4)	

Table 1: Scope of CC concepts

5 Security problem definition

5.1 Users

The users of an integrated product are:

- the entities who interact with the product through the physical or logical external interfaces of the Basic TOE or the IC.

The users of a layered product are of two kinds:

- the entities who interact with the product through the physical or logical external interfaces of the Extended TOE or the IC;
- the Application Layer that interacts with the product via the API of the Extended TOE.

5.2 Assets

Assets are of two kinds:

- valuable elements protected by the product itself (i.e. Security IC and TOE);
- product specifications, development tools and technology, and related documentation, protected by the product development environment.

The following list presents all the assets of the product together with the properties to be enforced.

Embedded Software code

The code of the software embedded in the product:

- o in integrated products, the asset stands for the code of the native OS and its applicative behaviour;
- o in layered products, the asset stands for the code of the native OS and possible applicative behaviour as well as the code of the Application Layer.

Properties: Correct execution, integrity and confidentiality.

Application note:

Both the IC and the TOE shall contribute to the correct execution and integrity of the embedded software code. The IC shall also protect the confidentiality of the asset.

Embedded Software data

The sensitive TSF and User data processed or stored by the product:

- o in integrated products, the asset stands for the data of the native OS and its applicative behaviour;
- o in layered products, the asset stands for the data of the native OS and possibly applicative behaviour as well as the data of the Application Layer.

Properties: Integrity and/or confidentiality.

Application note:

The confidentiality and the integrity of the sensitive data shall be ensured by a combination of Security IC and TOE mechanisms.

IC security services

The security services provided by the IC to the TOE, including the data needed to provide these services.

Properties: Integrity and correct execution.

Random numbers

Random numbers generated by the product.

Property: Unpredictability. The random numbers may be integer and/or confidential TSF data or User data, depending on their use.

Application note:

The protection required by the random numbers depends on their use. For instance, random numbers used to generate cryptographic keys shall be protected in confidentiality and integrity, while challenges used in an authentication protocol shall be protected in integrity only. The ST author shall indicate the usage of the random numbers generated by the product and the measures that are taken to protect them according to their usage.

5.3 Threats

This section presents the threats to the product during Phase 7 "Final Usage" of the product life cycle.

The threat agent or "attacker" is a hostile user (cf. section 5.1) that is physically and logically outside the product (hence outside the TOE).

5.3.1 ANSSI-CC-PP-ESforSSD_Basic - Integrated Product with Basic TOE

This section describes threats for integrated products with Basic TOE, applicable to ANSSI-CC-PP-ESforSSD_Basic.

T.Behaviour

An attacker modifies the behaviour of the product, e.g. by unauthorised use of commands (one or many incorrect commands, undefined commands, hidden commands) or buffer overflow attacks (overwriting buffer content to modify execution contexts or gaining system privileges).

An attacker performs perturbation attacks thus causing a malfunction of the product (for instance, by applying environmental stress) in order to deactivate or modify security features or functions of the product.

Related assets: IC security services, Embedded Software code.

T.Disclosure

An attacker discloses/accesses sensitive code or data by means of logical or physical attacks, for instance:

- brute force attacks;
- undocumented or undefined set of commands;
- input/output interfaces;
- access to residual data;
- clock frequency;
- power analysis (e.g. SPA, DPA).

Related assets: Embedded Software code and data. Random numbers for the purpose of key generation are a special case of such data.

T.Life_cycle

An attacker accesses to product functionalities outside of their expected availability range thus violating irreversible life cycle phases of the product (for instance, an attacker re-personalizes the product).

An attacker enters the Security IC test mode and uses the test features or interfaces to access or modify sensitive data or security features.

Related assets: IC security services, Embedded Software code and data, Random numbers.

T.Modification

An attacker modifies sensitive Embedded Software code or data by performing logical attacks (for instance, executing malicious code).

An attacker modifies sensitive Embedded Software code or data by performing perturbation attacks (for instance, modifying a value read from memory or changing the output of a computation whose result is written in memory).

An attacker modifies or disables security features of the product, using invasive attacks. These attacks may be performed using physical probing or physical manipulation of the hardware (reverse engineering, manipulation of memory cells, manipulation of hardware security parts).

Related assets: IC security services, Embedded Software code and data. Random numbers for any purpose (key generation and others such as challenges) are a special case of such data.

T.RND

An attacker predicts or obtains information about random numbers generated by the product. This may occur for instance by a lack of entropy of the random numbers generated by the product, or because the attacker forces the output of a predefined value.

Related assets: Random numbers.

5.3.2 ANSSI-CC-PP-ESforSSD_Extended - Layered Product with Extended TOE

The threats for layered products with Extended TOE, applicable to ANSSI-CC-PP-ESforSSD_Extended, consist of:

- all the threats for integrated products with Basic TOE defined in section 5.3.1, i.e. T.Disclosure, T.Modification, T.Behaviour, T.Life_cycle and T.RND;
- the following threat T.Separation.

T.Separation

An attacker tampers the product through the Application Layer thus bypassing the TOE API (services) to access or modify the Embedded Software code or data.

Related assets: Embedded Software code and data, random numbers.

5.4 Organisational Security Policies

The following OSP applies to any kind of product, with Basic or Extended TOE, i.e. it applies to the two Protection Profiles ANSSI-CC-PP-ESforSSD_Basic and ANSSI-CC-PP-ESforSSD_Extended.

OSP.Management_of_secrets

Management of secret data (e.g. generation, storage, distribution, destruction, loading into the product of cryptographic private keys, symmetric keys, user authentication data) performed outside the product on behalf of the TOE or Product Manufacturer shall comply with security organisational policies that enforce integrity and confidentiality of these data.

Secret data shared with the user of the product shall be exchanged through trusted channels that protect the data against unauthorised disclosure and modification and allow to detect potential security violations.

5.5 Assumptions

The following assumption concerns the product operational environment, after product delivery. It applies to any kind of product, with Basic or Extended TOE, that is, the assumption holds in the Protection Profiles ANSSI-CC-PP-ESforSSD_Basic and ANSSI-CC-PP-ESforSSD_Extended.

A.Protection_after_product_delivery

The product is assumed to be protected by the environment after delivery (may range from Phase 3 to 6) and before entering the final usage phase (Phase 7 of the life cycle). It is assumed that the persons manipulating the product in the operational environment follow the product guides (user and administrator guidance of the product, installation documentation and personalization guide). It is also assumed that the persons responsible for the application of the procedures contained in the guides, and the persons involved in delivery and protection of the product have the required skills and are aware of the security issues.

Note: The product certificate is valid only when the guides are applied. For instance, for pre-personalization or personalization guides, only the described set-up configurations or personalization profiles are covered by the certificate; any divergence would not be covered by the certificate.

6 Security Objectives

6.1 Security Objectives for the TOE

6.1.1 ANSSI-CC-PP-ESforSSD_Basic - Basic TOE

This section describes security objectives for the Basic TOE, applicable to ANSSI-CC-PP-ESforSSD_Basic.

O.Atomicity

The TOE shall provide a means to perform memory operations atomically.

O.Confidentiality

The TOE shall ensure that confidential information processed and stored by the TOE is protected against unauthorised disclosure.

O.Crypto

The TOE shall provide cryptographic services conformant to the cryptographic quality requirements specified in national schemes.

Application note:

In France, the requirements in [CRYPTO] apply.

O.Integrity

The TOE shall ensure that the Embedded Software code and data are protected against unauthorised modification.

O.Life_cycle

The TOE shall manage its own life cycle states as well as reversible and irreversible transitions between them. The TOE shall reject operations unexpected in its current life cycle.

O.Monitoring

The TOE shall monitor security registers and system flags made available to the IC and it shall respond to potential security violations in a way that preserves a secure state.

O.Operate

The TOE shall ensure the correct operation of its security functions, prevent the unauthorised use of commands and ensure that patches to the code are not bypassed.

O.RND

The TOE shall provide random numbers, resulting from an appropriate combination of hardware and/or software mechanisms, that are not predictable and that have sufficient entropy.

The TOE shall ensure that no information about the provided random numbers is available to an attacker since they might be used to generate cryptographic keys.

Random numbers shall be conformant to the quality requirements specified in national schemes.

Application note:

In France, the requirements in [CRYPTO] apply.

6.1.2 ANSSI-CC-PP-ESforSSD_Extended - Extended TOE

The objectives for the Extended TOE, applicable to ANSSI-CC-PP-ESforSSD_Extended, consist of:

- all the objectives for integrated products with Basic TOE defined in section 6.1.1, i.e. O.Confidentiality, O.Integrity, O.Operate, O.Monitoring, O.Atomicity, O.Life_cycle, O.RND, and O.Crypto;
- the following objective O.Separation.

O.Separation

The TOE shall provide a separation mechanism between itself and the Application Layer. The TOE shall control the access to its services and resources.

6.2 Security objectives for the Operational Environment

The following security objectives for the operational environment apply to any kind of product, integrated (with Basic TOE) and layered (with Extended TOE), i.e. they apply to the two Protection Profiles ANSSI-CC-PP-ESforSSD_Basic and ANSSI-CC-PP-ESforSSD_Extended.

6.2.1 Security objectives for the Security IC

OE.Physical

The Security IC shall detect and respond to invasive physical attacks, to environmental stress and to attempts to access Security IC unauthorised functionality. The Security IC shall prevent leakage of information. The Security IC shall manage its life cycle states and transitions between them; in particular the Security IC shall not allow Test Mode functions once the Security IC has entered the User Mode. The Security IC security features shall resist to high attack potential as defined in [CC AP].

A Security IC that complies with [PP0035] meets this objective.

OE.RND

The Security IC shall provide a random number generator conformant with the quality requirements specified in national schemes.

Random numbers output by this generator shall not be predictable and shall have sufficient entropy.

The Security IC shall ensure that no information about the produced random numbers is available to an attacker since they might be used to generate cryptographic keys.

Application note:

In France, the requirements in [CRYPTO] apply.

6.2.2 Security objectives from product delivery up to Phase 7

OE.Management_of_Secrets

The secret User or TSF data managed outside the TOE shall be protected against unauthorised disclosure and modification.

OE.Protection_After_Product_Delivery

Procedures and controlled environment shall ensure protection of the product and related information after delivery. Procedures shall ensure that people involved in product delivery and protection have the required skills. The persons using the product in the operational environment shall apply the product guides (user and administrator guidance of the product, installation documentation and personalization guide).

6.3 Security Objectives Rationale

6.3.1 Threats

6.3.1.1 ANSSI-CC-PP-ESforSSD_Basic - Integrated Product with Basic TOE

T.Behaviour OE.Physical requires protection against physical attacks corrupting the execution of the Embedded Software code. O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could indicate execution perturbation. O.Operate requires the correct execution of Embedded Software code and ensures that only correct commands shall be processed and that patches are applied, if any. O.Atomicity prevents reaching inconsistent states through the interruption of memory operations.

The fulfillment of all these objectives allows to remove the threat.

T.Disclosure OE.Physical requires preventing the leakage of Embedded Software code and data by the IC. It covers the hardware aspects of the threat. The objective O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could indicate information leakage. O.Confidentiality requires the protection of confidential data (Embedded Software TSF or User data) from unauthorised disclosure by the TOE. O.Crypto requires cryptographic capacities from the TOE, which can be used to enforce data confidentiality.

The fulfillment of all these objectives allows to remove the threat.

T.Life_cycle OE.Physical and O.Life_cycle require the control the IC and TOE life cycles, respectively, to prevent abuse of functionality by physical and logical means.

The fulfillment of all these objectives allows to remove the threat.

T.Modification OE.Physical requires ensuring the integrity of the Embedded Software code and data by the IC. It covers the hardware aspects of the threat. The objective O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could indicate information modification. O.Integrity requires the

protection of integer data (Embedded Software TSF or User data) from unauthorised modification by the TOE. O.Crypto requires cryptographic capacities from the TOE that can be used to enforce data integrity.

The fulfillment of all these objectives allows to remove the threat.

T.RND OE.RND requires a random number generator from the IC that meets national standards and O.RND requires providing random numbers by an appropriate combination of hardware and software mechanisms that also meets national quality metrics. OE.Physical ensures the protection of IC security features, including random number generation. O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could have an impact on the quality of random numbers generated by the IC.

The fulfillment of all these objectives allows to remove the threat.

6.3.1.2 ANSSI-CC-PP-ESforSSD_Extended - Layered Product with Extended TOE

The rationales between threats and objectives of ANSSI-CC-PP-ESforSSD_Basic given in section 6.3.1.1 fully apply to ANSSI-CC-PP-ESforSSD_Extended. The objective O.Separation contributes to each of them and to the threat T.Separation as specified below (for each threat, the full rationale is given).

T.Behaviour OE.Physical requires protection against physical attacks corrupting the execution of the Embedded Software code. O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could indicate execution perturbation. O.Operate requires the correct execution of Embedded Software code and ensures that only correct commands shall be processed and that patches are applied, if any. O.Atomicity prevents reaching inconsistent states through the interruption of memory operations.

O.Separation requires protecting the execution of the TOE from the Application Layer.

The fulfillment of all these objectives allows to remove the threat.

T.Disclosure OE.Physical requires preventing the leakage of Embedded Software code and data by the IC. It covers the hardware aspects of the threat. The objective O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could indicate information leakage. O.Confidentiality requires the protection of confidential data (Embedded Software TSF or User data) from unauthorised disclosure by the TOE. O.Crypto requires cryptographic capacities from the TOE, which can be used to enforce data confidentiality.

O.Separation requires protecting the confidentiality of the TOE resources, including TSF and User data, from the Application Layer.

The fulfillment of all these objectives allows to remove the threat.

T.Life_cycle OE.Physical and O.Life_cycle require the control the IC and TOE life cycles, respectively, to prevent abuse of functionality by physical and logical means.

O.Separation requires protecting the life cycle of the TOE from the Application Layer.

The fulfillment of all these objectives allows to remove the threat.

T.Modification OE.Physical requires ensuring the integrity of the Embedded Software code and data by the IC. It covers the hardware aspects of the threat. The objective

O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could indicate information modification. O.Integrity requires the protection of integer data (Embedded Software TSF or User data) from unauthorised modification by the TOE. O.Crypto requires cryptographic capacities from the TOE that can be used to enforce data integrity.

O.Separation requires protecting the integrity of the TOE resources, including TSF and User data, from the Application Layer.

The fulfillment of all these objectives allows to remove the threat.

T.RND OE.RND requires a random number generator from the IC that meets national standards and O.RND requires providing random numbers by an appropriate combination of hardware and software mechanisms that also meets national quality metrics. OE.Physical ensures the protection of IC security features, including random number generation. O.Monitoring ensures that the TOE shall react to any security alert made available by the IC, including those that could have an impact on the quality of random numbers generated by the IC.

O.Separation requires that the TOE protects the random number functionality from the Application Layer.

The fulfillment of all these objectives allows to remove the threat.

T.Separation O.Separation directly covers the threat.

The fulfillment of this objective allows to remove the threat.

6.3.2 Organisational Security Policies

OSP.Management_of_secrets OE.Management_of_Secrets directly covers the organisational security policy.

6.3.3 Assumptions

A.Protection_after_product_delivery OE.Protection_After_Product_Delivery directly covers the assumption.

6.3.4 SPD and Security Objectives

6.3.4.1 ANSSI-CC-PP-ESforSSD_Basic – Integrated Product with Basic TOE

The following tables show the relationship between the elements in the SPD for integrated products (threats, assumptions and organisational security policies) and the objectives for the Basic TOE and for the operational environment.

Threats	Security Objectives	Rationale
T.Behaviour	O.Atomicity, O.Monitoring, O.Operate, OE.Physical	Section 6.3.1
T.Disclosure	O.Confidentiality, O.Crypto, O.Monitoring, OE.Physical	Section 6.3.1
T.Life_cycle	O.Life_cycle, OE.Physical	Section 6.3.1
T.Modification	O.Crypto, O.Integrity, O.Monitoring, OE.Physical	Section 6.3.1
T.RND	O.Monitoring, O.RND, OE.Physical, OE.RND	Section 6.3.1

Table 2 Threats and Security Objectives – ANSSI-CC-PP-ESforSSD_Basic

Security Objectives	Threats
O.Atomicity	T.Behaviour
O.Confidentiality	T.Disclosure
O.Crypto	T.Disclosure, T.Modification
O.Integrity	T.Modification
O.Life_cycle	T.Life_cycle
O.Monitoring	T.Behaviour,T.Disclosure, T.Modification, T.RND
O.Operate	T.Behaviour
O.RND	T.RND
OE.Physical	T.Behaviour, T.Disclosure, T.Life_cycle, T.Modification, T.RND
OE.RND	T.RND
OE.Management_of_Secrets	
OE.Protection_After_Product_Delivery	

Table 3 Security Objectives and Threats – ANSSI-CC-PP-ESforSSD_Basic

Organisational Security Policies	Security Objectives	Rationale
OSP.Management_of_secrets	OE.Management_of_Secrets	Section 6.3.2

Table 4 OSPs and Security Objectives - ANSSI-CC-PP-ESforSSD_Basic

Security Objectives	Organisational Security Policies
O.Atomicity	
O.Confidentiality	
O.Crypto	
O.Integrity	
O.Life_cycle	
O.Monitoring	
O.Operate	
O.RND	
OE.Physical	
OE.RND	
OE.Management_of_Secrets	OSP.Management_of_secrets
OE.Protection_After_Product_Delivery	

Table 5 Security Objectives and OSPs – ANSSI-CC-PP-ESforSSD_Basic

Assumptions	Security objectives for the Operational Environment	Rationale
A.Protection_after_product_delivery	OE.Protection_After_Product_Delivery	Section 6.3.3

Table 6 Assumptions and Security Objectives for the Operational Environment – ANSSI-CC-PP-ESforSSD_Basic

Security objectives for the Operational Environment	Assumptions
OE.Physical	
OE.RND	
OE.Management_of_Secrets	
OE.Protection_After_Product_Delivery	A.Protection_after_product_delivery

Table 7 Security Objectives for the Operational Environment and Assumptions - ANSSI-CC-PP-ESforSSD_Basic

6.3.4.2 ANSSI-CC-PP-ESforSSD_Extended – Layered Product with Extended TOE

The following tables show the relationship between the SPD (threats, assumptions, OSP) and the objectives for layered products. The only difference with the tables in section 6.3.4.1 for integrated products is the presence of T.Separation and O.Separation.

Threats	Security Objectives	Rationale
T.Behaviour	O.Atomicity, O.Monitoring, O.Operate, O.Separation, OE.Physical	Section 6.3.1
T.Disclosure	O.Confidentiality, O.Crypto, O.Monitoring, O.Separation, OE.Physical	Section 6.3.1
T.Life_cycle	O.Life_cycle, O.Separation, OE.Physical	Section 6.3.1
T.Modification	O.Crypto, O.Integrity, O.Monitoring, O.Separation, OE.Physical	Section 6.3.1
T.RND	O.Monitoring, O.RND, OE.Physical, O.Separation, OE.RND	Section 6.3.1
T.Separation	O.Separation	Section 6.3.1

Table 8 Threats and Security Objectives – ANSSI-CC-PP-ESforSSD_Extended

Security Objectives	Threats
O.Atomicity	T.Behaviour
O.Confidentiality	T.Disclosure
O.Crypto	T.Disclosure, T.Modification
O.Integrity	T.Modification
O.Life_cycle	T.Life_cycle
O.Monitoring	T.Behaviour,T.Disclosure, T.Modification, T.RND
O.Operate	T.Behaviour
O.RND	T.RND
O.Separation	T.Behaviour, T.Disclosure, T.Life_cycle, T.Modification, T.RND, T.Separation
OE.Physical	T.Behaviour, T.Disclosure, T.Life_cycle, T.Modification, T.RND
OE.RND	T.RND
OE.Management_of_Secrets	
OE.Protection_After_Product_Delivery	

Table 9 Security Objectives and Threats – ANSSI-CC-PP-ESforSSD_Extended

Organisational Security Policies	Security Objectives	Rationale
OSP.Management_of_secrets	OE.Management_of_Secrets	Section 6.3.2

Table 10 OSPs and Security Objectives – ANSSI-CC-PP-ESforSSD_Extended

Security Objectives	Organisational Security Policies
O.Atomicity	
O.Confidentiality	
O.Crypto	
O.Integrity	
O.Life_cycle	
O.Monitoring	
O.Operate	
O.RND	
O.Separation	
OE.Physical	
OE.RND	
OE.Management_of_Secrets	OSP.Management_of_secrets
OE.Protection_After_Product_Delivery	

Table 11 Security Objectives and OSPs – ANSSI-CC-PP-ESforSSD_Extended

Assumptions	Security objectives for the Operational Environment	Rationale
A.Protection_after_product_delivery	OE.Protection_After_Product_Delivery	Section 6.3.3

Table 12 Assumptions and Security Objectives for the Operational Environment – ANSSI-CC-PP-ESforSSD_Extended

Security objectives for the Operational Environment	Assumptions
OE.Physical	
OE.RND	
OE.Management_of_Secrets	
OE.Protection_After_Product_Delivery	A.Protection_after_product_delivery

Table 13 Security Objectives for the Operational Environment and Assumptions - ANSSI-CC-PP-ESforSSD_Extended

7 Security Requirements

7.1 Security Functional Requirements

This section states the security functional requirements for the TOE.

7.1.1 ANSSI-CC-PP-ESforSSD_Basic - Basic TOE

This section presents the security functional requirements for the Basic TOE, applicable to ANSSI-CC-PP-ESforSSD_Basic.

7.1.1.1 Atomicity

The Security Functional Requirements of this section support the objective O.Atomicity. They address the rollback of incomplete memory writings upon software or hardware interruption.

The goal of SFRs is the following:

- definition of the operations for which rollback is allowed, through FDP_ACC.1;
- rollback rules through FDP_ROL.1.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author is allowed to add product dependent requirements, for instance complementary FDP_ACF.1 rules or iterations of FDP_ROL.1 to address roll forward operations.

FDP_ACC.1 / Atomicity Subset access control

FDP_ACC.1.1 / Atomicity The TSF shall enforce the **Access Control Policy for Atomicity** on **[assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]**.

Application note:

The conditions under which the operations can be rolled back are stated in FDP_ROL.1/Atomicity.

FDP_ROL.1 / Atomicity Basic rollback

FDP_ROL.1.1 / Atomicity The TSF shall enforce **Access Control Policy for Atomicity** to permit the rollback of the **[assignment: list of operations]** on the **[assignment: information and / or list of objects]**.

FDP_ROL.1.2 / Atomicity The TSF shall permit operations to be rolled back within the **[assignment: boundary limit to which rollback may be performed]**.

Application note:

The ST author is allowed to iterate this requirement to address operations other than rollback that contribute to the same atomicity purpose, e.g. roll forward upon delete operations.

7.1.1.2 Confidentiality

The Security Functional Requirements of this section support the objective O.Confidentiality. They address confidential TSF and User data during processing and transfer as well as when data is stored in memories.

The goal of the SFRs is the following:

- access control through FDP_ACC.1 and FDP_ACF.1: any confidential TSF or User data must be protected against unauthorised access. Depending on the kind of data and their persistence, access control may translate for instance, into credentials for granting access or into encryption rules. These SFRs apply both to User and TSF confidential data. The requirements FMT_MSA.1 and FMT_MSA.3 allow to specify the management of the security attributes;
- protected communication of confidential through FDP_UCT.1 (User data) and through FPT_ITC.1 (TSF data). Depending on the data and on the characteristics of the environment, the communication rules may imply, for instance, encryption of confidential data or strong authentication of the receptor;
- no residual confidential information available through FDP_RIP.1. This SFR applies to TSF and User confidential data without distinction;
- protection of confidential data during processing through FDP_UNO.1. This SFR applies to TSF and User confidential data without distinction.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author is allowed to add product dependent requirements, for instance import/export rules by means of FDP_ITC/FDP_ETC.

FDP_ACC.1 / Confidentiality Subset access control

FDP_ACC.1.1 / Confidentiality The TSF shall enforce the **Access Control Policy for Confidentiality** on **[assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]**.

Refinement:

"Objects" stands for TSF and User confidential data.

Application note:

Subjects are active entities within the TOE performing operations on behalf of a user (each particular applicative behaviour could be modeled by a subject). The TSF itself is a particular subject since this SFR applies both to User and TSF confidential data.

Objects are TSF and User confidential data, which depend on the functionalities of the TOE (e.g. keys, pins, etc).

Operations stand for access operations to confidential data. These operations depend on the kind of data (e.g. read/write, cipher/decipher) and include communication from and to the TOE.

FDP_ACF.1/Confidentiality Security attribute based access control

FDP_ACF.1.1/Confidentiality The TSF shall enforce the **Access Control Policy for Confidentiality** to objects based on the following: **[assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]**.

FDP_ACF.1.2/Confidentiality The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]**.

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

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

Application note:

Security attributes depend on the actual subjects and objects defined in FDP_ACC.1/Confidentiality.

Rules specify the access policy implemented by the TOE. For instance:

- rules that grant access based on credentials of the subject. These credentials can be modeled as security attributes of the subject;
- rules that impose encryption/decryption of specific objects. The subject must be allowed to use the expected key.

FDP_RIP.1/Confidentiality Subset residual information protection

FDP_RIP.1.1/Confidentiality The TSF shall ensure that any previous information content of a resource is made unavailable upon the **[selection: allocation of the resource to,**

deallocation of the resource from] the following objects: **[assignment: list of objects]**.

Refinement:

"Information content" stands for TSF and User confidential data.

FDP_UCT.1/Confidentiality Basic data exchange confidentiality

FDP_UCT.1.1/Confidentiality The TSF shall enforce the **Access Control Policy for Confidentiality** to **[selection: transmit, receive]** user data in a manner protected from unauthorised disclosure.

Refinement:

"User data" stands for confidential User data.

FMT_MSA.1/Confidentiality Management of security attributes

FMT_MSA.1.1/Confidentiality The TSF shall enforce the **Access Control Policy for Confidentiality** to restrict the ability to **[selection: change_default, query, modify, delete, [assignment: other operations]]** the security attributes **[assignment: list of security attributes]** to **[assignment: the authorised identified roles]**.

Application note:

The security attributes belong to the set of attributes defined in FDP_ACF.1/Confidentiality.

FMT_MSA.3/Confidentiality Static attribute initialisation

FMT_MSA.3.1/Confidentiality The TSF shall enforce the **Access Control Policy for Confidentiality** to provide **[selection: choose one of: restrictive, permissive, [assignment: other property]]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/Confidentiality The TSF shall allow the **[assignment: the authorised identified roles]** to specify alternative initial values to override the default values when an object or information is created.

FPR_UNO.1/Confidentiality Unobservability

FPR_UNO.1.1/Confidentiality The TSF shall ensure that **[assignment: list of users and/or subjects]** are unable to observe the operation **[assignment: list of operations]** on **[assignment: list of objects]** by **[assignment: list of protected users and/or subjects]**.

Application note:

"Observe the operation" stands for "observe the processing linked to the operation in a way that could allow disclosing confidential information". The operations concerned are those that involve cryptographic keys or any other confidential data.

FPT_ITC.1/Confidentiality Inter-TSF confidentiality during transmission

FPT_ITC.1.1/Confidentiality The TSF shall protect all TSF data transmitted from the TSF to another trusted IT product from unauthorised disclosure during transmission.

Refinement:

"TSF data" stands for TSF data with confidentiality constraints.

7.1.1.3 Cryptography

The Security Functional Requirements of this section support the objective O.Crypto, as well as O.Disclosure and O.Integrity. They address the cryptographic functionalities of the TSF. This PP assumes that the TOE implements at least one cryptographic function.

The goal of the SFRs is the following:

- specification of the cryptographic operations implemented by the TOE through FCS_COP.1;
- specification of the key destruction mechanisms through FCS_CKM.4.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author shall add either FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2 to specify the way cryptographic keys enter the product. Moreover, the ST author is allowed to iterate these requirements as many times as needed.

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 **[assignment: cryptographic key destruction method]** that meets the following: **[assignment: list of standards]**.

Application note:

The ST author shall choose the standards according to the requirements of the national scheme (for instance [CRYPTO] in France).

FCS_COP.1 Cryptographic operation

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

Application note:

The ST author shall choose the standards according to requirements of the national scheme (for instance [CRYPTO] in France).

7.1.1.4 Integrity

The Security Functional Requirements of this section support the objective O.Integrity. They address TSF and User data with integrity constraints (referred to as "integer data" in the following) during processing, transfer and storage in memories. They address also TSF code.

The goal of the SFRs is the following:

- access control through FDP_ACC.1 and FDP_ACF.1: integer data must be protected against unauthorised modification. Depending on the kind of data and their persistence, access control may translate for instance, into credentials for granting modification or into cryptographic rules that mandate integrity checking before using the data. These SFRs apply both to User and TSF integer data. The requirements FMT_MSA.1 and FMT_MSA.3 allow to specify the management of the security attributes;
- protected communication of integer data through FDP_UIT.1 (User data) and FPT_ITI.1 (TSF data). Depending on the data and on the characteristics of the environment, the communication rules may imply, for instance, the use of cryptographic mechanisms or strong authentication of the receptor;
- integrity monitoring through FDP_SDI.2 (applied both to TSF and User integer data) and FPT_TST.1 (applied to TSF and User data as well as to TSF code).

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author is allowed to add product dependent requirements, for instance import/export rules by means of FDP_ITC/FDP_ETC.

FDP_ACC.1 / Integrity Subset access control

FDP_ACC.1.1 / Integrity The TSF shall enforce the **Access Control Policy for Integrity** on **[assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]**.

Refinement:

"Objects" stands for TSF and User integer data.

Application note:

Subjects are active entities within the TOE performing operations on behalf of a user (each particular applicative behaviour could be modeled by a subject). The TSF itself is a particular subject since this SFR applies both to User and TSF integer data.

Objects are TSF and User integer data (e.g. keys, pins, counters).

Operations stand for operations that modify integer data (e.g. update key, change pin, increment counter).

FDP_ACF.1/Integrity Security attribute based access control

FDP_ACF.1.1/Integrity The TSF shall enforce the **Access Control Policy for Integrity** to objects based on the following: **[assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]**.

FDP_ACF.1.2/Integrity The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]**.

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

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

Application note:

Security attributes depend on the actual subjects and objects defined in FDP_ACC.1/Integrity.

Rules specify the access policy implemented by the TOE. For instance:

- rules that grant modification based on credentials of the subject. These credentials can be modeled as security attributes of the subject;
- rules that impose the use of integrity cryptographic mechanisms to specific objects. The subject must be allowed to use the necessary cryptographic elements.

FDP_SDI.2/Integrity Stored data integrity monitoring and action

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

FDP_SDI.2.2/Integrity Upon detection of a data integrity error, the TSF shall **[assignment: action to be taken]**.

Refinement:

"User data" shall be understood as integer TSF and User data.

Application note:

The developer shall provide the list of integer TSF and User data that are monitored, which must be consistent with the objects defined in the Access Control Policy for Integrity.

FDP_UIT.1 / Integrity Data exchange integrity

FDP_UIT.1.1 / Integrity The TSF shall enforce the **Access Control Policy for Integrity** to **[selection: transmit, receive]** user data in a manner protected from **[selection: modification, deletion, insertion, replay]** errors.

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

Refinement:

"User data" stands for User data with integrity constraints.

FMT_MSA.1 / Integrity Management of security attributes

FMT_MSA.1.1 / Integrity The TSF shall enforce the **Access Control Policy for Integrity** to restrict the ability to **[selection: change_default, query, modify, delete, [assignment: other operations]]** the security attributes **[assignment: list of security attributes]** to **[assignment: the authorised identified roles]**.

FMT_MSA.3 / Integrity Static attribute initialisation

FMT_MSA.3.1 / Integrity The TSF shall enforce the **Access Control Policy for Integrity** to provide **[selection: choose one of: restrictive, permissive, [assignment: other property]]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 / Integrity The TSF shall allow the **[assignment: the authorised identified roles]** to specify alternative initial values to override the default values when an object or information is created.

FPT_ITI.1 / Integrity Inter-TSF detection of modification

FPT_ITI.1.1 / Integrity The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and another trusted IT product within the following metric: **[assignment: a defined modification metric]**.

FPT_ITI.1.2 / Integrity The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and another trusted IT product and perform **[assignment: action to be taken]** if modifications are detected.

Refinement:

"TSF data" stands for TSF data with integrity constrains.

7.1.1.5 Life cycle

The Security Functional Requirements of this section support the objective O.Life_cycle. They address the control of the life cycle states of the TSF and the transitions between them.

The goal of the SFRs is the definition of the states of the life cycle and access control rules to the operations in each state through FDP_ACC.1 and FDP_ACF.1. The transitions between states are among the operations. The requirements FMT_MSA.1 and FMT_MSA.3 allow to specify the management of the security attributes.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. Note that the TOE may support many coexisting life cycles, for instance, one corresponding to each applicative behaviour, that is, one for each integrated application. The ST author is allowed to iterate these SFRs as many times as needed according to the product characteristics.

FDP_ACC.1 / Life_cycle Subset access control

FDP_ACC.1.1 / Life_cycle The TSF shall enforce the **Access Control Policy for Life Cycle** on **[assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]**.

Application note:

States can be modeled through subjects and objects. Operations stand either for transitions between states or any other kind of command allowed in the states of the life cycle.

FDP_ACF.1/Life_cycle Security attribute based access control

FDP_ACF.1.1/Life_cycle The TSF shall enforce the **Access Control Policy for Life Cycle** to objects based on the following: **[assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]**.

FDP_ACF.1.2/Life_cycle The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]**.

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

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

FMT_MSA.1/Life_cycle Management of security attributes

FMT_MSA.1.1/Life_cycle The TSF shall enforce the **Access Control Policy for Life Cycle** to restrict the ability to **[selection: change_default, query, modify, delete, [assignment: other operations]]** the security attributes **[assignment: list of security attributes]** to **[assignment: the authorised identified roles]**.

FMT_MSA.3/Life_cycle Static attribute initialisation

FMT_MSA.3.1/Life_cycle The TSF shall enforce the **Access Control Policy for Life Cycle** to provide **[selection: choose one of: restrictive, permissive, [assignment: other property]]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/Life_cycle The TSF shall allow the **[assignment: the authorised identified roles]** to specify alternative initial values to override the default values when an object or information is created.

7.1.1.6 Monitoring

The Security Functional Requirements of this section support the objective O.Monitoring. They address the monitoring of the potential security violations detected by the underlying IC.

The goal of the SFRs is the detection and response to potential security violations through FAU_ARP.1 and FAU_SAA.1. For each of the audited events, the developer shall provide the response implemented by the TSF.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author is allowed to add product dependent requirements, for instance complementary Security Audit FAU requirements or FPT_RCV "Trusted Recovery".

FAU_ARP.1 / Monitoring Security alarms

FAU_ARP.1.1 / Monitoring The TSF shall take **[assignment: list of actions]** upon detection of a potential security violation.

Application note:

The ST author shall specify for each kind of potential security violation the list of associated actions.

FAU_SAA.1 / Monitoring Potential violation analysis

FAU_SAA.1.1 / Monitoring The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the enforcement of the SFRs.

FAU_SAA.1.2 / Monitoring The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of **[assignment: subset of defined auditable events]** known to indicate a potential security violation;

b) **[assignment: any other rules]**.

Refinement:

The "audited events" are detected by the TSF or by the underlying IC. In particular, the TSF shall monitor all the events generated by the Security IC physical detectors that are made available to the TSF (e.g. by interrupt routines or status flags).

7.1.1.7 Operate

The Security Functional Requirements of this section support the objective O.Operate. They address the correct execution of the TSF.

The goal of the SFRs is the following:

- testing of critical functionalities and detection of integrity errors in TSF data or executable code through FPT_TST.1;
- secure state in case of failure through FPT_FLS.1;
- controlled activation, deactivation and configuration of functionality and data through FMT_MOF.1 and FMT_MTD.1.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author is allowed to add product dependent requirements, for instance FPT_TDC.1 for interpretation rules applicable to TSF data or FMT_SMF.1 for the specification of security management functions.

FMT_MOF.1/Operate Management of security functions behaviour

FMT_MOF.1.1/Operate The TSF shall restrict the ability to **[selection: determine the behaviour of, disable, enable, modify the behaviour of]** the functions **[assignment: list of functions]** to **[assignment: the authorised identified roles]**.

Refinement:

The patches of the TOE, uniquely referenced in the Configuration List, are excluded from the list of functions that can be enabled or disabled: the TOE patches are necessarily enabled. Instead, the functionalities implemented by these patches can be configured, if applicable.

FMT_MTD.1/Operate Management of TSF data

FMT_MTD.1.1/Operate The TSF shall restrict the ability to **[selection: change_default, query, modify, delete, clear, [assignment: other operations]]** the **[assignment: list of TSF data]** to **[assignment: the authorised identified roles]**.

FPT_FLS.1/Operate Failure with preservation of secure state

FPT_FLS.1.1/Operate The TSF shall preserve a secure state when the following types of failures occur: **[assignment: list of types of failures in the TSF]**.

Application note:

The list of failures includes those failures resulting from potential security violation detected by FAU_SAA.1/Monitoring.

FPT_TST.1/Operate TSF testing

FPT_TST.1.1/Operate The TSF shall run a suite of self tests **[selection: during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which self test should occur]]** to demonstrate the correct operation of **[selection: [assignment: parts of TSF], the TSF]**.

FPT_TST.1.2/Operate The TSF shall provide authorised users with the capability to verify the integrity of **[selection: [assignment: parts of TSF data], TSF data]**.

FPT_TST.1.3/Operate The TSF shall provide authorised users with the capability to verify the integrity of **[selection: [assignment: parts of TSF], TSF]**.

Application note:

The TOE can totally or partially rely on the underlying IC to fulfill this requirement. The ST author shall indicate the solution implemented by the product under evaluation.

7.1.1.8 Random numbers

The Security Functional Requirements of this section support the objective O.RND.

The goal of the FIA_SOS.2 requirement is to state the characteristics of the random numbers provided by the TOE.

FIA_SOS.2/RND TSF Generation of secrets

FIA_SOS.2.1/RND The TSF shall provide a mechanism to generate secrets that meet **[assignment: random numbers quality metric]**.

FIA_SOS.2.2/RND The TSF shall be able to enforce the use of TSF generated secrets for **[assignment: list of TSF functions]**.

Refinement:

The term "secrets" stands for "random numbers".

Application note:

The "quality metric" shall meet national schemes requirements (e.g. [CRYPTO] in France). The random number generator is an appropriate combination of IC/TOE mechanisms. It is possible that the underlying IC already provides random numbers meeting the requirements and that the TOE does not supplement it.

The ST author shall indicate the requirements and/or standards that are applied and the functions that use the generated random numbers (authentication operations, key generation, etc).

7.1.1.9 Roles

The Security Functional Requirements of this section support the objectives O.Atomicity, O.Confidentiality, O.Integrity, O.Life_cycle and O.Operate.

The goal of the SFRs is as follows:

- definition of the roles involved in the management of the security attributes (FMT_MSA.1 and FMT_MSA.3), of the security functions (FMT_MOF.1) and of TSF data (FMT_MTD.1);
- specification of the functionality of the TOE before user identification by means of FIA_UID.1.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author is allowed to add product dependent requirements, for instance, the definition of the specific security management functions by means of FMT_SMF.1. This could be necessary, for instance, if the standard CC operations on security attributes - change_default, query, modify, delete (cf. FMT_MSA.1) - are not appropriate.

FIA_UID.1 Timing of identification

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

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

Application note:

The ST author shall indicate how the user is identified. He/she can use FIA requirements for this purpose.

FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles **[assignment: the authorised identified roles]**.

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

7.1.2 ANSSI-CC-PP-ESforSSD_Extended - Extended TOE

The security functional requirements for the Extended TOE applicable to ANSSI-CC-PP-ESforSSD_Extended consist of:

- the security functional requirements for integrated products with Basic TOE defined in section 7.1.1;
- the requirements on the separation between the Application Layer and the Extended TOE presented hereafter.

7.1.2.1 Separation

The Security Functional Requirements of this section support the objective O.Separation. They address the protection of the TSF from the Application Layer and the interaction between these two entities through the TOE API exclusively.

The goal of the SFRs is to control the information flow between the TSF and the Application Layer through FDP_IFC.1 and FDP_IFF.1 The Application Layer can only access the TSF through the TOE API. The requirements FMT_MSA.1 and FMT_MSA.3 are CC dependencies linked to the information flow control policy.

This is a minimum set of SFRs, for which the ST author shall provide a complete instantiation. The ST author is allowed to add product dependent requirements.

FDP_IFC.1/Separation Subset information flow control

FDP_IFC.1.1/Separation The TSF shall enforce the **Information Flow Control Policy for Application Layer Separation** on **[assignment: list of subjects, information,**

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

Application note:

The subjects involved in the policy are the Application Layer and the TSF itself. Information flows between them through operations available from the TOE API exclusively.

FDP_IFF.1/Separation Simple security attributes

FDP_IFF.1.1/Separation The TSF shall enforce the **Information Flow Control Policy for Application Layer Separation** based on the following types of subject and information security attributes: **[assignment: list of subjects and information controlled under the indicated SFP, and for each, the security attributes]**.

FDP_IFF.1.2/Separation The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **[assignment: for each operation, the security attribute-based relationship that must hold between subject and information security attributes]**.

FDP_IFF.1.3/Separation The TSF shall enforce the **[assignment: additional information flow control SFP rules]**.

FDP_IFF.1.4/Separation The TSF shall explicitly authorise an information flow based on the following rules: **[assignment: rules, based on security attributes, that explicitly authorise information flows]**.

FDP_IFF.1.5/Separation The TSF shall explicitly deny an information flow based on the following rules: **[assignment: rules, based on security attributes, that explicitly deny information flows]**.

FMT_MSA.3/Separation Static attribute initialisation

FMT_MSA.3.1/Separation The TSF shall enforce the **Information Flow Control Policy for Application Layer Separation** to provide **[selection: choose one of: restrictive, permissive, [assignment: other property]]** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/Separation The TSF shall allow the **[assignment: the authorised identified roles]** to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.1/Separation Management of security attributes

FMT_MSA.1.1/Separation The TSF shall enforce the **Information Flow Control Policy for Application Layer Separation** to restrict the ability to **[selection: change_default, query, modify, delete, [assignment: other operations]]** the security attributes **[assignment: list of security attributes]** to **[assignment: the authorised identified roles]**.

7.2 Security Assurance Requirements

The security assurance requirement level is EAL4 augmented with AVA_VAN.5 and ALC_DVS.2.

7.3 Security Requirements Rationale

7.3.1 Objectives

7.3.1.1 ANSSI-CC-PP-ESforSSD_Basic - Basic TOE

O.Atomicity FDP_ACC.1/Atomicity defines the memory operations for which rollback (or any other repairing operation defined by the ST author) is allowed.

FDP_ROL.1/Atomicity states the rules for rollback that leaves the TOE in a secure state.

O.Confidentiality FDP_ACF.1/Confidentiality, FDP_ACC.1/Confidentiality, FMT_MSA.1/Confidentiality, FMT_MSA.3/Confidentiality and FMT_SMR.1 define the security policy enforcing access control of confidential TSF and User data (the "Access Control Policy for Confidentiality"). These requirements specify the data that is protected by the policy (i.e. the "objects"), the active entities that can potentially access the data (the "subjects"), the kinds of access covered by the policy (the "operations"), the access rules to objects based on the characteristics (the "security attributes") of the subjects and objects, the management of these characteristics by the authorised roles. The authorised entities are those subjects that satisfy the conditions imposed by the access rules. FIA_UID.1 states the operations allowed before the user enrolls a legitimate role.

FDP_UCT.1/Confidentiality on User data and FPT_ITC.1/Confidentiality on TSF data provides minimum requirements for protected communication of confidential data: the operations of transmission and/or reception of confidential data must satisfy the constraints of the "Access Control Policy for Confidentiality".

FCS_COP.1 and FCS_CKM.4 specifies the characteristics of cryptographic operations and key destruction methods the Access Control Policy for Confidentiality relies on.

FPR_UNO.1/Confidentiality enforces the unobservability of confidential TSF or User data processing.

FDP_RIP.1/Confidentiality ensures that no residual confidential TSF or User data is available.

O.Crypto FCS_COP.1 and FCS_CKM.4 specify the cryptographic operations and the key destruction mechanisms implemented by the TOE, according to requirements of the national schemes.

O.Integrity FDP_ACF.1/Integrity, FDP_ACC.1/Integrity, FMT_MSA.1/Integrity, FMT_MSA.3/Integrity and FMT_SMR.1 define the security policy enforcing access control of integer TSF and User data (the "Access Control Policy for Integrity"). These requirements specify the data that is protected by the policy (i.e. the "objects"), the active entities that can potentially access the data (the "subjects"), the kinds of access covered by the policy (the "operations"), the access rules to objects based on the characteristics (the "security attributes") of the subjects and objects, the management of these characteristics by the authorised roles. The authorised entities are those subjects that satisfy the conditions imposed by the access rules. FIA_UID.1 states the operations allowed before the user enrolls a legitimate role.

FDP_UIT.1/Integrity on User data and FPT_ITI.1/Integrity on TSF data provides minimum requirements for protected communication of integer data: the operations of transmission and/or reception of confidential data must satisfy the constraints of the "Access Control Policy for Integrity".

FCS_COP.1 and FCS_CKM.4 specifies the characteristics of cryptographic operations and key destruction methods the Access Control Policy for Integrity relies on.

FDP_SDI.2/Integrity specifies the TSF and User data that is monitored and the response of the TOE. These data belongs to the "objects" specified in the Access Control Policy for Integrity.

FPT_TST.1/Operate contributes to the integrity protection of TSF and User data as well as TSF code.

O.Life_cycle FDP_ACF.1/Life_cycle, FDP_ACC.1/Life_cycle, FMT_MSA.1/Life_cycle, FMT_MSA.3/Life_cycle and FMT_SMR.1 define the security policy for the management of the TOE life cycle (the "Access Control Policy for life cycle"). These requirements specify the states of the life cycle (i.e. the "objects"), the active entities (the "subjects") that can potentially perform life cycle transitions (the "operations"), the transition rules based on the characteristics (the "security attributes") of the subjects and states, the management of these characteristics by the authorised roles. The authorised entities are those subjects that satisfy the conditions imposed by the transition rules. FIA_UID.1 states the transitions allowed before the user enrolls a legitimate role, if any.

O.Monitoring FAU_SAA.1/Monitoring specifies the events that are audited for detection of potential security violations made available by the IC.

FAU_ARP.1/Monitoring requires that the TSF responds to potential security violations.

O.Operate FPT_TST.1/Operate contributes to the correct execution of TSF code through the testing of critical functionalities and detection of integrity errors in TSF data or executable code.

FPT_FLS.1/Operate requires that failures do not impact on the security of the TOE.

FMT_MOF.1/Operate requires controlled activation, deactivation and configuration of functionality, performed by authorised roles. Moreover, this requirement does require that all patches of the TOE are executed.

FMT_MTD.1/Operate requires controlled access to TSF data, allowed only to authorised roles.

FMT_SMR.1 and FIA_UID.1 states the authorised roles and the actions allowed to the user before he/she enrolls any role.

O.RND FIA_SOS.2/RND requires the TOE to provide a random number generation mechanism that meets the national schemes requirements.

7.3.1.2 ANSSI-CC-PP-ESforSSD_Extended - Extended TOE

The rationales between objectives and SFRs of ANSSI-CC-PP-ESforSSD_Basic fully apply to ANSSI-CC-PP-ESforSSD_Extended without any addition, we refer to section 7.3.1.1.

The rationale concerning O.Separation is given below.

O.Separation FDP_IFC.1/Separation, FDP_IFF.1/Separation, FMT_MSA.3/Separation and FMT_MSA.1/Separation state the policy for controlling the access to TOE services and resources by the Application Layer ("Information Flow Control Policy for Application Layer Separation") The Application Layer can only access the TSF through the TOE API.

FMT_SMR.1 defines the roles authorised to manage the attributes introduced in FMT_MSA.1/Separation and FMT_MSA.3/Separation and FIA_UID.1 lists the actions that do not require any particular role.

7.3.2 Rationale tables of Security Objectives and SFRs

7.3.2.1 ANSSI-CC-PP-ESforSSD_Basic - Basic TOE

The following tables show the relationship between the objectives for the Basic TOE and the SFRs.

Security Objectives	Security Functional Requirements	Rationale
O.Atomicity	FDP_ROL.1/Atomicity, FDP_ACC.1/Atomicity	Section 7.3.1
O.Confidentiality	FDP_RIP.1/Confidentiality, FDP_ACF.1/Confidentiality, FMT_MSA.1/Confidentiality, FMT_MSA.3/Confidentiality, FDP_ACC.1/Confidentiality, FMT_SMR.1, FPR_UNO.1/Confidentiality, FDP_UCT.1/Confidentiality, FPT_ITC.1/Confidentiality, FIA_UID.1, FCS_COP.1, FCS_CKM.4	Section 7.3.1
O.Crypto	FCS_CKM.4, FCS_COP.1	Section 7.3.1
O.Integrity	FDP_SDI.2/Integrity, FDP_ACC.1/Integrity, FDP_ACF.1/Integrity, FMT_MSA.3/Integrity, FMT_MSA.1/Integrity, FDP_UIT.1/Integrity, FPT_ITI.1/Integrity, FMT_SMR.1, FIA_UID.1, FCS_COP.1, FCS_CKM.4, FPT_TST.1/Operate	Section 7.3.1
O.Life_cycle	FDP_ACC.1/Life_cycle, FDP_ACF.1/Life_cycle, FMT_MSA.3/Life_cycle, FMT_MSA.1/Life_cycle, FMT_SMR.1, FIA_UID.1	Section 7.3.1
O.Monitoring	FAU_ARP.1/Monitoring, FAU_SAA.1/Monitoring	Section 7.3.1
O.Operate	FMT_MOF.1/Operate, FPT_TST.1/Operate, FMT_SMR.1, FIA_UID.1, FMT_MTD.1/Operate, FPT_FLS.1/Operate	Section 7.3.1
O.RND	FIA_SOS.2/RND	Section 7.3.1

Table 14 Security Objectives and SFRs – ANSSI-CC-PP-ESforSSD_Basic

Security Functional Requirements	Security Objectives
FDP_ACC.1/Atomicity	O.Atomicity
FDP_ROL.1/Atomicity	O.Atomicity
FDP_ACC.1/Confidentiality	O.Confidentiality
FDP_ACF.1/Confidentiality	O.Confidentiality
FDP_RIP.1/Confidentiality	O.Confidentiality
FDP_UCT.1/Confidentiality	O.Confidentiality
FMT_MSA.1/Confidentiality	O.Confidentiality
FMT_MSA.3/Confidentiality	O.Confidentiality
FPR_UNO.1/Confidentiality	O.Confidentiality
FPT_ITC.1/Confidentiality	O.Confidentiality
FCS_CKM.4	O.Confidentiality, O.Integrity, O.Crypto
FCS_COP.1	O.Confidentiality, O.Integrity, O.Crypto
FDP_ACC.1/Integrity	O.Integrity
FDP_ACF.1/Integrity	O.Integrity
FDP_SDI.2/Integrity	O.Integrity
FDP_UIT.1/Integrity	O.Integrity
FMT_MSA.1/Integrity	O.Integrity
FMT_MSA.3/Integrity	O.Integrity
FPT_ITI.1/Integrity	O.Integrity
FDP_ACC.1/Life_cycle	O.Life_cycle
FDP_ACF.1/Life_cycle	O.Life_cycle
FMT_MSA.1/Life_cycle	O.Life_cycle
FMT_MSA.3/Life_cycle	O.Life_cycle
FAU_ARP.1/Monitoring	O.Monitoring
FAU_SAA.1/Monitoring	O.Monitoring
FMT_MOF.1/Operate	O.Operate
FMT_MTD.1/Operate	O.Operate
FPT_FLS.1/Operate	O.Operate
FPT_TST.1/Operate	O.Integrity, O.Operate
FIA_SOS.2/RND	O.RND
FIA_UID.1	O.Confidentiality, O.Integrity, O.Operate, O.Life_cycle
FMT_SMR.1	O.Confidentiality, O.Integrity, O.Operate, O.Life_cycle

Table 15 SFRs and Security Objectives – ANSSI-CC-PP-ESforSSD_Basic

7.3.2.2 ANSSI-CC-PP-ESforSSD_Extended - Extended TOE

The following tables show the relationship between the objectives for the Extended TOE and the SFRs. The only difference with the tables in section 7.3.2.1 is the presence of O.Separation and the related SFRs.

Security Objectives	Security Functional Requirements	Rationale
O.Atomicity	FDP_ROL.1/Atomicity, FDP_ACC.1/Atomicity	Section 7.3.1
O.Confidentiality	FDP_RIP.1/Confidentiality, FDP_ACF.1/Confidentiality, FMT_MSA.1/Confidentiality, FMT_MSA.3/Confidentiality, FDP_ACC.1/Confidentiality, FMT_SMR.1, FPR_UNO.1/Confidentiality, FDP_UCT.1/Confidentiality, FPT_ITC.1/Confidentiality, FIA_UID.1, FCS_COP.1, FCS_CKM.4	Section 7.3.1
O.Crypto	FCS_CKM.4, FCS_COP.1	Section 7.3.1
O.Integrity	FDP_SDI.2/Integrity, FDP_ACC.1/Integrity, FDP_ACF.1/Integrity, FMT_MSA.3/Integrity, FMT_MSA.1/Integrity, FDP_UIT.1/Integrity, FPT_ITI.1/Integrity, FMT_SMR.1, FIA_UID.1, FCS_COP.1, FCS_CKM.4, FPT_TST.1/Operate	Section 7.3.1
O.Life_cycle	FDP_ACC.1/Life_cycle, FDP_ACF.1/Life_cycle, FMT_MSA.3/Life_cycle, FMT_MSA.1/Life_cycle, FMT_SMR.1, FIA_UID.1	Section 7.3.1
O.Monitoring	FAU_ARP.1/Monitoring, FAU_SAA.1/Monitoring	Section 7.3.1
O.Operate	FMT_MOF.1/Operate, FPT_TST.1/Operate, FMT_SMR.1, FIA_UID.1, FMT_MTD.1/Operate, FPT_FLS.1/Operate	Section 7.3.1
O.RND	FIA_SOS.2/RND	Section 7.3.1
O.Separation	FMT_SMR.1, FDP_IFC.1/Separation, FDP_IFF.1/Separation, FMT_MSA.3/Separation, FMT_MSA.1/Separation, FIA_UID.1	Section 7.3.1

Table 16 Security Objectives and SFRs – ANSSI-CC-PP-ESforSSD_Extended

Security Functional Requirements	Security Objectives
FDP_ACC.1/Atomicity	O.Atomicity
FDP_ROL.1/Atomicity	O.Atomicity
FDP_ACC.1/Confidentiality	O.Confidentiality
FDP_ACF.1/Confidentiality	O.Confidentiality
FDP_RIP.1/Confidentiality	O.Confidentiality
FDP_UCT.1/Confidentiality	O.Confidentiality
FMT_MSA.1/Confidentiality	O.Confidentiality
FMT_MSA.3/Confidentiality	O.Confidentiality
FPR_UNO.1/Confidentiality	O.Confidentiality
FPT_ITC.1/Confidentiality	O.Confidentiality
FCS_CKM.4	O.Confidentiality, O.Integrity, O.Crypto
FCS_COP.1	O.Confidentiality, O.Integrity, O.Crypto
FDP_ACC.1/Integrity	O.Integrity
FDP_ACF.1/Integrity	O.Integrity
FDP_SDI.2/Integrity	O.Integrity
FDP_UIT.1/Integrity	O.Integrity
FMT_MSA.1/Integrity	O.Integrity
FMT_MSA.3/Integrity	O.Integrity
FPT_ITI.1/Integrity	O.Integrity
FDP_ACC.1/Life_cycle	O.Life_cycle
FDP_ACF.1/Life_cycle	O.Life_cycle
FMT_MSA.1/Life_cycle	O.Life_cycle
FMT_MSA.3/Life_cycle	O.Life_cycle
FAU_ARP.1/Monitoring	O.Monitoring
FAU_SAA.1/Monitoring	O.Monitoring
FMT_MOF.1/Operate	O.Operate
FMT_MTD.1/Operate	O.Operate
FPT_FLS.1/Operate	O.Operate
FPT_TST.1/Operate	O.Integrity, O.Operate
FIA_SOS.2/RND	O.RND
FIA_UID.1	O.Confidentiality, O.Integrity, O.Operate, O.Life_cycle, O.Separation
FMT_SMR.1	O.Confidentiality, O.Integrity, O.Operate, O.Life_cycle, O.Separation

Security Functional Requirements	Security Objectives
FDP_IFC.1/Separation	O.Separation
FDP_IFF.1/Separation	O.Separation
FMT_MSA.3/Separation	O.Separation
FMT_MSA.1/Separation	O.Separation

Table 17 SFRs and Security Objectives – ANSSI-CC-PP-ESforSSD_Extended

7.3.3 Dependencies

7.3.3.1 SFRs dependencies

Requirements	CC Dependencies	Satisfied Dependencies
FDP_ACC.1/Atomicity	(FDP_ACF.1)	
FDP_ROL.1/Atomicity	(FDP_ACC.1 or FDP_IFC.1)	FDP_ACC.1/Atomicity
FDP_ACC.1/Confidentiality	(FDP_ACF.1)	FDP_ACF.1/Confidentiality
FDP_ACF.1/Confidentiality	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.1/Confidentiality, FMT_MSA.3/Confidentiality
FDP_RIP.1/Confidentiality	No dependencies	
FDP_UCT.1/Confidentiality	(FDP_ACC.1 or FDP_IFC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_ACC.1/Confidentiality
FMT_MSA.1/Confidentiality	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.1/Confidentiality, FMT_SMR.1
FMT_MSA.3/Confidentiality	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/Confidentiality, FMT_SMR.1
FPR_UNO.1/Confidentiality	No dependencies	
FPT_ITC.1/Confidentiality	No dependencies	
FCS_CKM.4	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2)	
FCS_COP.1	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	FCS_CKM.4
FDP_ACC.1/Integrity	(FDP_ACF.1)	FDP_ACF.1/Integrity
FDP_ACF.1/Integrity	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.1/Integrity, FMT_MSA.3/Integrity
FDP_SDI.2/Integrity	No dependencies	
FDP_UIT.1/Integrity	(FDP_ACC.1 or FDP_IFC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_ACC.1/Integrity
FMT_MSA.1/Integrity	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.1/Integrity, FMT_SMR.1

Requirements	CC Dependencies	Satisfied Dependencies
FMT_MSA.3/Integrity	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/Integrity, FMT_SMR.1
FPT_ITI.1/Integrity	No dependencies	
FDP_ACC.1/Life_cycle	(FDP_ACF.1)	FDP_ACF.1/Life_cycle
FDP_ACF.1/Life_cycle	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.1/Life_cycle, FMT_MSA.3/Life_cycle
FMT_MSA.1/Life_cycle	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.1/Life_cycle, FMT_SMR.1
FMT_MSA.3/Life_cycle	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/Life_cycle, FMT_SMR.1
FAU_ARP.1/Monitoring	(FAU_SAA.1)	FAU_SAA.1/Monitoring
FAU_SAA.1/Monitoring	(FAU_GEN.1)	
FMT_MOF.1/Operate	(FMT_SMF.1) and (FMT_SMR.1)	FMT_SMR.1
FMT_MTD.1/Operate	(FMT_SMF.1) and (FMT_SMR.1)	FMT_SMR.1
FPT_FLS.1/Operate	No dependencies	
FPT_TST.1/Operate	No dependencies	
FIA_SOS.2/RND	No dependencies	
FIA_UID.1	No dependencies	
FMT_SMR.1	(FIA_UID.1)	FIA_UID.1
FDP_IFC.1/Separation	(FDP_IFF.1)	FDP_IFF.1/Separation
FDP_IFF.1/Separation	(FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.1/Separation, FMT_MSA.3/Separation
FMT_MSA.3/Separation	(FMT_MSA.1) and (FMT_SMR.1)	FMT_SMR.1, FMT_MSA.1/Separation
FMT_MSA.1/Separation	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FMT_SMR.1, FDP_IFC.1/Separation

Table 18 SFRs dependencies

Rationale for the exclusion of dependencies

The dependency FTP_ITC.1 or FTP_TRP.1 of FDP_UCT.1/Confidentiality is unsupported. The requirement FDP_UTC.1 allows to state the property that confidential data shall enter and leave the TOE according to the Access Control Policy for

Confidentiality. This PP does not mandate any TOE capacity to establish trusted paths or trusted channels.

The dependency FMT_SMF.1 of FMT_MSA.1/Confidentiality is unsupported. This PP does not mandate any specific security management function. The ST author shall add FMT_SMF if necessary (for instance, if the standard CC operation for the management of security attributes are not enough or appropriate).

The dependency FMT_SMF.1 of FMT_MSA.1/Integrity is unsupported. This PP does not mandate any specific security management function. The ST author shall add FMT_SMF if necessary (for instance, if the standard CC operations for the management of security attributes are not enough or appropriate).

The dependency FTP_ITC.1 or FTP_TRP.1 of FDP_UIT.1/Integrity is unsupported. The requirement FDP_UIT.1 allows to state the property that integer data shall enter and leave the TOE according to the Access Control Policy for Integrity. This PP does not mandate any TOE capacity to establish trusted paths or trusted channels.

The dependency FMT_SMF.1 of FMT_MOF.1/Operate is unsupported. This PP does not mandate any particular security management function.

The dependency FMT_SMF.1 of FMT_MTD.1/Operate is unsupported. This PP does not mandate any particular security management function.

The dependency FAU_GEN.1 of FAU_SAA.1/Monitoring is unsupported. This PP does not mandate any particular audit record generation.

The dependency FDP_ACF.1 of FDP_ACC.1/Atomicity is unsupported. The FDP_ACC requirement serves as a framework for the definition of the operations that can be rolled back. There is no need to require FDP_ACF since there is neither mandatory attribute nor rule.

The dependency FMT_SMF.1 of FMT_MSA.1/Life_cycle is unsupported. This PP does not mandate any specific security management function. The ST author shall add FMT_SMF if necessary (for instance, if the standard CC operations for the management of security attributes are not enough or appropriate).

The dependency FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2 of FCS_COP.1 is unsupported. This PP does not choose between key generation and import. The ST author shall add either FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2 to specify the way each key becomes available for cryptographic operations.

The dependency FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2 of FCS_CKM.4 is unsupported. This PP does not choose between key generation and import. The ST author shall add either FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2 to specify the way each cryptographic key becomes available for cryptographic operations.

The dependency FMT_SMF.1 of FMT_MSA.1/Separation is unsupported. This PP does not mandate any specific security management function. The ST author shall add FMT_SMF if necessary (for instance, if the standard CC operations for the management of security attributes are not enough or appropriate).

7.3.3.2 SARs dependencies

Requirements	CC Dependencies	Satisfied Dependencies
ADV_ARC.1	(ADV_FSP.1) and (ADV_TDS.1)	ADV_FSP.4, ADV_TDS.3
ADV_FSP.4	(ADV_TDS.1)	ADV_TDS.3
ADV_IMP.1	(ADV_TDS.3) and (ALC_TAT.1)	ADV_TDS.3, ALC_TAT.1
ADV_TDS.3	(ADV_FSP.4)	ADV_FSP.4
AGD_OPE.1	(ADV_FSP.1)	ADV_FSP.4
AGD_PRE.1	No dependencies	
ALC_CMC.4	(ALC_CMS.1) and (ALC_DVS.1) and (ALC_LCD.1)	ALC_CMS.4, ALC_DVS.2, ALC_LCD.1
ALC_CMS.4	No dependencies	
ALC_DEL.1	No dependencies	
ALC_DVS.2	No dependencies	
ALC_LCD.1	No dependencies	
ALC_TAT.1	(ADV_IMP.1)	ADV_IMP.1
ASE_CCL.1	(ASE_ECD.1) and (ASE_INT.1) and (ASE_REQ.1)	ASE_ECD.1, ASE_INT.1, ASE_REQ.2
ASE_ECD.1	No dependencies	
ASE_INT.1	No dependencies	
ASE_OBJ.2	(ASE_SPD.1)	ASE_SPD.1
ASE_REQ.2	(ASE_ECD.1) and (ASE_OBJ.2)	ASE_ECD.1, ASE_OBJ.2
ASE_SPD.1	No dependencies	
ASE_TSS.1	(ADV_FSP.1) and (ASE_INT.1) and (ASE_REQ.1)	ADV_FSP.4, ASE_INT.1, ASE_REQ.2
ATE_COV.2	(ADV_FSP.2) and (ATE_FUN.1)	ADV_FSP.4, ATE_FUN.1
ATE_DPT.1	(ADV_ARC.1) and (ADV_TDS.2) and (ATE_FUN.1)	ADV_ARC.1, ADV_TDS.3, ATE_FUN.1
ATE_FUN.1	(ATE_COV.1)	ATE_COV.2
ATE_IND.2	(ADV_FSP.2) and (AGD_OPE.1) and (AGD_PRE.1) and (ATE_COV.1) and (ATE_FUN.1)	ADV_FSP.4, AGD_OPE.1, AGD_PRE.1, ATE_COV.2, ATE_FUN.1

Requirements	CC Dependencies	Satisfied Dependencies
AVA_VAN.5	(ADV_ARC.1) and (ADV_FSP.4) and (ADV_IMP.1) and (ADV_TDS.3) and (AGD_OPE.1) and (AGD_PRE.1) and (ATE_DPT.1)	ADV_ARC.1, ADV_FSP.4, ADV_IMP.1, ADV_TDS.3, AGD_OPE.1, AGD_PRE.1, ATE_DPT.1

Table 19 SARs dependencies

7.3.4 Rationale for the Security Assurance Requirements

EAL4 is required for this type of TOE and product since they are meant to resist against sophisticated attacks. This evaluation assurance level allows a developer to gain maximum assurance from positive security engineering based on good practices. EAL4 represents the highest practical level of assurance expected for a commercial grade product. In order to provide a meaningful level of assurance that the TOE and its embedding product provide an adequate level of defense against such attacks, the evaluators must have access to the low level design and source code. The lowest for which such access is required is EAL4.

7.3.5 AVA_VAN.5 Advanced methodical vulnerability analysis

Due to the definition of the TOE and of the embedding the product, the product must resist to high attack potential. This is due to the fact that the product (Smart Secure Device) can be placed in a hostile environment, such as electronic laboratories. This robustness level is achieved by the assurance AVA_VAN.5 component. Independent vulnerability analysis is based on highly detailed technical information. The attacker is assumed to be thoroughly familiar with the specific implementation of the TOE. The attacker is presumed to have a high level of technical knowledge. AVA_VAN.5 has dependencies with ADV_ARC.1, ADV_FSP.4, ADV_TDS.3, ADV_IMP.1, AGD_PRE.1, AGD_OPE.1 and ATE_DPT.1. All these dependencies are satisfied by EAL4.

7.3.6 ALC_DVS.2 Sufficiency of security measures

Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE and the embedding product. This assurance component is a higher hierarchical component to EAL4 (only ALC_DVS.1 is found in EAL4). Due to the nature of the TOE and embedding product, there is a need to justify the sufficiency of these procedures to protect their confidentiality and integrity. ALC_DVS.2 has no dependencies.

Index