LUNA® PCI CONFIGURED FOR USE IN LUNA® SA 4.5.1 (RF) WITH BACKUP SECURITY TARGET

DOCUMENT NUMBER:	CR-3636
AUTHOR:	Terry Fletcher
DEPARTMENT:	Engineering
LOCATION OF ISSUE:	Ottawa
DATE ORIGINATED:	January 19, 2012
REVISION LEVEL:	5
REVISION DATE:	July 24, 2013
SUPERSESSION DATA:	CR-3636, Revision 4 dated January 28, 2013
SECURITY LEVEL:	Non-sensitive

DOCUMENT CHANGE HISTORY

Revision	Date	Reason for Change	Sections Affected
Original	January 19, 2012	First release of document	All
1	February 19, 2012	Minor edits.	All
2	October 23, 2012	Updates incorporated based on lab feedback.	Section 6.2.1,
3	December 12, 2012	Updated to provide guidelines on RPED.	2.6.1, 5.2.4.1, 6.2.1, 6.3.15
4	January 28, 2013	Updated to reflect SA software version change to 4.5.1 and provide guidelines on RPED.	1.1, 1.2, 2.6.1, 5.1.4.3, 5.2.4.1, 6.2.1
5	July 24, 2013	Update to provide additional guidance regarding RPED.	2.6.1, 6.2.1, 6.3.15 Figure 2

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY OF TERMS[1]

Administrator means a CSP user role that performs TOE initialisation or other TOE administrative functions. These tasks are mapped to the Security Officer role of the TOE.

Advanced electronic signature means an electronic signature which meets the following requirements:

(a) it is uniquely linked to the signatory;

(b) it is capable of identifying the signatory;

(c) it is created using means that the signatory can maintain under his sole control, and

(d) it is linked to the data to which it relates in such a manner that any subsequent change of the data is detectable.

Approved algorithms and parameters means cryptographic algorithms and parameters approved for use in electronic signatures, secure signature creation devices and trustworthy systems under the Directive or by the appropriate national standards body.

Authentication data is information used to verify the claimed identity of a user.

Auditor means a user exporting the TOE audit data and reviewing the audit data with tools in the TOE environment.

Backup means secure export and external storage of the CSP-SCD, the TSF data and the system data (backup data) sufficient to recreate the state of the TOE at the time the backup was created. Note that backup is the only function which is allowed to export CSP-SCD.

CEN European Committee for Standardization (CEN). The Protection Profiles used as the basis for this Security Target were developed through CEN Workshop Agreements (CWA) under the European Electronic Signature Standardisation Initiative (EESSI) CEN/ISSS electronic signature (E-SIGN) workshop, Area F on secure signature-creation devices (SSCD) and Area D2 on trustworthy systems.

Certificate means an electronic attestation which links the SVD to a person and confirms the identity of that person.

CSP signature creation data (CSP-SCD) means SCD which is used by the CSP, e.g. for the creation of advanced electronic signatures in qualified certificates or for signing certificate status information.

CSP signature verification data (CSP-SVD) means SVD which corresponds to the CSP-SCD and which is used to verify the advanced electronic signature in the qualified certificate.

Certification Authority (CA) is a synonym for CSP, defined below.

Certification-Service-provider (CSP) means an entity or a legal or natural person who issues certificates or provides other services related to electronic signatures.

Data to be signed (DTBS) means the complete electronic data to be signed, such as QC content data or certificate status information.

Data to be signed representation (DTBS-representation) means the data sent to the TOE for signing and is

(a) a hash-value of the DTBS or

(b) an intermediate hash-value of a first part of the DTBS and a remaining part of the DTBS or

(c) the DTBS itself.

The client indicates to the TOE the case of DTBS-representation, unless implicitly indicated. The hash-value in case (a) or the intermediate hash-value in case (b) is calculated by the client. The final hash-value in case (b) or the hash-value in case (c) is calculated by the TOE.

[1] This Glossary of Terms is based on the Glossaries provided in the SSCD PP and the CMCSOB PP.

Directive The Directive 1999/93/ec of the European parliament and of the council of 13 December 1999 on a Community framework for electronic signatures [1] is also referred to as the 'Directive' in the remainder of the ST.

Digital signature means data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery e.g. by the recipient. [ISO 7498-2]

Dual person control means a special form of access control of a task which requires two users with different identities to be authenticated and authorised to the defined roles at the time this task is to be performed.

Hardware security module (HSM) means a cryptographic module used to generate the advanced signature in qualified certificates. The TOE specified in this Security Target is an HSM.

Peripheral Component Interconnect (PCI) is a data bus standard used in most PCs and servers.

Qualified certificate means a certificate which meets the requirements laid down in Annex I of the Directive and is provided by a CSP who fulfils the requirements laid down in Annex II of the Directive.

Reference authentication data (RAD) means data persistently stored by the TOE for verification of the authentication attempt as authorised user. Note that the TOE does not store such data.

Restore means import of the backup data to recreate the state of the TOE at the time the backup was created.

Secure signature-creation device (SSCD) means configured software or hardware which is used to implement the SCD and which meets the requirements laid down in Annex III of the Directive.

Side-channel means illicit information flow resulting from observation of the physical behavior of the technical implementation of the TOE. Side-channels are limited to interfaces not intended for data output like power consumption, timing of any signals and radiation. Side-channels might be enhanced by influencing the TOE behavior from outside.

Signature-creation data (SCD) means unique data, such as codes or private cryptographic keys, which are used by the signatory to create an electronic signature.

Signature-verification data (SVD) means data, such as codes or public cryptographic keys, which are used for the purpose of verifying an electronic signature. (defined in the Directive [1], article 2.7)

SSCD provision service means a service that prepares and provides a SSCD to subscribers.

User means any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

User data means data created by and for the user that does not affect the operation of the TSF.

Verification authentication data (VAD) means authentication data provided as input by knowledge or authentication data derived from user's biometric characteristics.

LIST OF ACRONYMS AND ABBREVIATIONS

API	Application Programming Interface
CA	Certificate Authority
CC	Common Criteria
CIMS	Certificate Issuing and Management System
CLI	Command Line Interface
CMCSO(B)	Cryptographic Module for Certificate Signing Operations (with Backup)
COTS	Commercial Off-the-Shelf
CSP	Certification Service Provider
DES	Data Encryption Standard
DLL	Dynamic Linked Library
DSA	Digital Signature Algorithm
DTBS	Data to be Signed
EAL	Evaluation Assurance Level
FIPS	Federal Information Processing Standard
HSM	Hardware Security Module
IETF	Internet Engineering Task Force
IT	Information Technology
IT	Information Technology
PC	Personal Computer
PCI	Peripheral Component Interconnect
PED	PIN Entry Device
PIN	Personal Identification Number
PKC	Public Key Confirmation
PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
PP	Protection Profile
PRNG	Pseudo-Random Number Generator
RAD	Reference Authentication Data
RAM	Random Access Memory
RF	Refresh
RNG	Random Number Generator (Generation)
ROM	Read-Only Memory
RSA	Asymmetric algorithm developed by Rivest, Shamir and Adleman
SAR	Security Assurance Requirements
SCD	Signature Creation Data
SF	Security Function
SFP	Security Function Policy
SFR	Security Functional Requirements
SHA	Secure HASH Algorithm
SO	Security Officer
SoF	Strength of Function
SSCD	Secure Signature Creation Device
ST	Security Target
SVD	Signature Verification Data
TDES	Triple DES
TOE	Target of Evaluation
TSA	Time Stamp Authority
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSFI	TSF Interface
TSP	TOE Security Policy
UAV	User Authorization Vector
VAD	Verification Authentication Data

Document Organisation

A **Glossary of Terms and a List of Acronyms and Abbreviations** list is provided to define frequently used terms and acronyms.

Section 1 provides the introductory material for the Security Target.

Section 2 provides general purpose and TOE description.

Section 3 provides a discussion of the expected environment for the TOE. This section also defines the set of threats that are to be addressed by either the technical countermeasures implemented in the TOE hardware, the TOE software, or through the environmental controls.

Section 4 defines the security objectives for both the TOE and the TOE environment.

Section 5 contains the functional requirements and assurance requirements derived from the Common Criteria (CC), Part 2 [2] and Part 3 [3], that must be satisfied by the TOE.

Section 6 provides the TOE Summary Specification.

Section 7 contains the Protection Profile conformance claims.

Section 8 provides the rationale to explicitly demonstrate that the information technology security objectives satisfy the policies and threats. Arguments are provided for the coverage of each policy and threat. The section then explains how the set of requirements are complete relative to the objectives, and that each security objective is addressed by one or more component requirements. Arguments are provided for the coverage of each objective. Next section 8 provides a set of arguments that address dependency analysis, strength of function issues, and the internal consistency and mutual supportiveness of the protection profile requirements

Appendix A contains the list of references used in compiling this ST.

1. ST INTRODUCTION

1.1. ST Identification

Title:	Luna® PCI Configured for Use in Luna® SA 4.5.1 (RF) With Backup Security Target
TOE Name	Luna® PCI Configured for Use in Luna SA 4.5.1 (RF)
TOE Version	Luna PCI Hardware Version VBD-03-0100 (part numbers 216-010031-001 [legacy part number 900691-000] and 216-010031-002 [legacy part number 900691-001]), Firmware Version 4.8.7
Assurance level:	EAL 4-augmented by ADV_IMP.2, ALC_FLR.2, AVA_CCA.1, AVA_MSU.3, AVA_VLA.4
Keywords:	Commercial-off-the-shelf (COTS), hardware security module, certification authority, certification service provider, key management, cryptographic services, key generation, key protection, digital certificate management, public-key infrastructure, digital signature, encryption, confidentiality, integrity, networked information systems, baseline information protection.

1.2. ST Overview

The Target of Evaluation (TOE) described by this Security Target (ST) is an implementation of the certified Protection Profile – CWA 14167-2 [Crypto Module for Certificate Signing Operations with Backup (CMCSOB) PP, version 0.28 dated 27 October 2003] published in the Official Journal of the European Commission in May 2004 as a generally recognized standard for electronic signature products. The TOE also implements security functions in addition to those specified in the CMCSOB PP.

The TOE, Luna® PCI configured for use in the Luna SA 4.5.1 (RF), includes the following:

- the Luna® PCI cryptographic module in a PCI Card form factor (216-010031-001 [legacy part number 900691-000] and 216-010031-002 [legacy part number 900691-001] with Firmware Version 4.8.7),
- a Luna® PIN Entry Device (PED) (Local PED – Firmware Versions 2.0.2 and 2.4.0-3) and iKeys,
- API library and driver software (version 4.5.1),
- Luna SA 4.5 / 4.5.1 Guidance Documentation (700-010478-002, Revision B).

The TOE Security Functions are all implemented within the Luna® PCI cryptographic module. The Luna® PCI cryptographic module is a Hardware Security Module (HSM) in the form of a PCI card that typically resides within a custom computing or secure communications appliance. It is contained in its own secure enclosure that provides physical resistance to tampering and zeroization of plaintext key material and security parameters in the event a tamper signal is received. The boundary of the cryptographic module is defined to encompass all components inside the secure enclosure on the PCI card.

SafeNet's Luna® HSMs comprise a range of hardware security solutions for digital identity applications. Luna products feature true hardware key management to maintain the confidentiality and integrity of digital signature and encryption keys. Key material is generated, stored, and used exclusively within the secure confines of the Luna hardware security module (HSM) to prevent compromise.

SafeNet's HSMs provide advanced features like direct hardware-to-hardware backup, split user role administration, multi-person authentication, and trusted path authentication coupled with proven security and operational deployment experience in some of the largest PKI deployments in the world.

1.3. CC Conformance Claim

This ST is conformant with:

1) CC Version 2.3 Part 2- extended. The following non Part 2 Security Functional Requirements are included to meet specific requirements of the TOE:

- FCS_RND.1 (Quality metrics for random numbers)
- FDP_BKP.1 (CMCSO PP Backup)

2) CC Version 2.3 Part 3-EAL 4 augmented. The EAL 4 package has been augmented by the addition of the following Part 3 requirements:

- ADV_IMP.2 (Implementation of the TSF)
- ALC_FLR.2 (Flaw Reporting Procedures)
- AVA_CCA.1 (Covert channel analysis)
- AVA_MSU.3 (Analysis and testing for insecure states)
- AVA_VLA.4 (Highly resistant)

2. TOE DESCRIPTION

The TOE provides a physically and logically protected component for the performance of cryptographic functions for key generation, key storage, encryption and decryption, digital signature and verification used by application systems that provide cryptographic support functions such as a Certificate Authority/Certification Service Provider (CA/CSP) or Time Stamp Authority (TSA). It includes processors, read-only and random-access memory, and firmware packaged in a tamper-resistant form along with Cryptographic API software that resides on the host computer.

Figure 1 shows the TOE in its appliance deployment configuration – as part of the Luna® SA network-attached appliance.

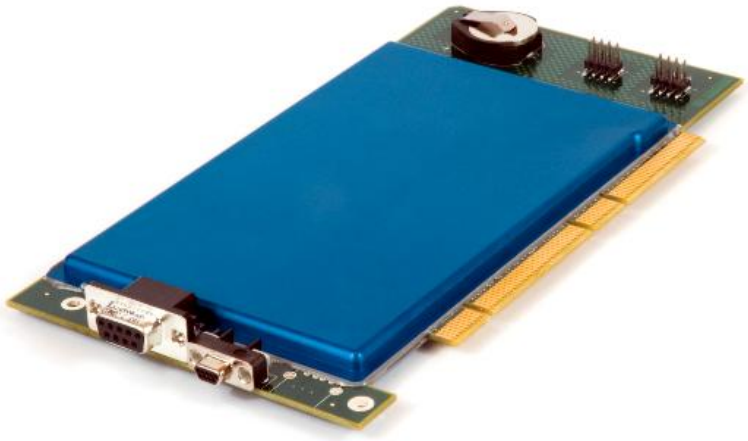

Figure 1. Luna PCI Cryptographic Module

Figure 2 Luna SA with Local PED and iKeys

The boundary of the TOE described in this ST encompasses the following:

1. The Luna® PCI cryptographic module – a printed circuit board in PCI card format enclosed within tamper-resistant metal covers. The printed circuit board hosts volatile and non-volatile memory, a microprocessor, with its associated firmware, data, control and key transfer signal paths, an FPGA that provides an entropy selection function for the on-board random bit generator, input/output controller, power management and a local oscillator.
2. The Luna® PIN Entry Device (Local PED), which is housed in a separate physical enclosure and, through a physically and electrically separate data port connection to the module, provides a trusted path for the communication of critical security parameters (authentication data and plaintext cryptographic parameters) to and from the module.
3. iKeys, which are USB token devices used to securely store authentication data and other critical security parameters for entry through the PED.
4. PKCS #11 client library and driver software provides the programming and communications interface normally used to access the cryptographic module.
5. User and Administrative Guidance documentation for the TOE is provided on CD-ROM along with client PKCS #11 software.

The TSF boundary is the Luna® PCI cryptographic module.

The TOE in the evaluated configuration is supported on the CentOS 5.4 operating system.

The TOE supports backup and restoration of cryptographic objects, such as the CSP-SCD, with the TSF data needed to re-establish an operational state after recovery from a failure. Backup and restoration is done using cryptographic protocols and mechanisms that protect the confidentiality of the backup data and detect loss of the integrity of the backup data. Measures must be taken within the non-IT environment to ensure the availability of the backup data.

Figure 3 below illustrates the relationship between the TOE and the Luna SA appliance and also shows the division between the TSF and non-TSF portions of the TOE, specifically the Luna PCI HSM and the supporting Luna PED and library and driver software.

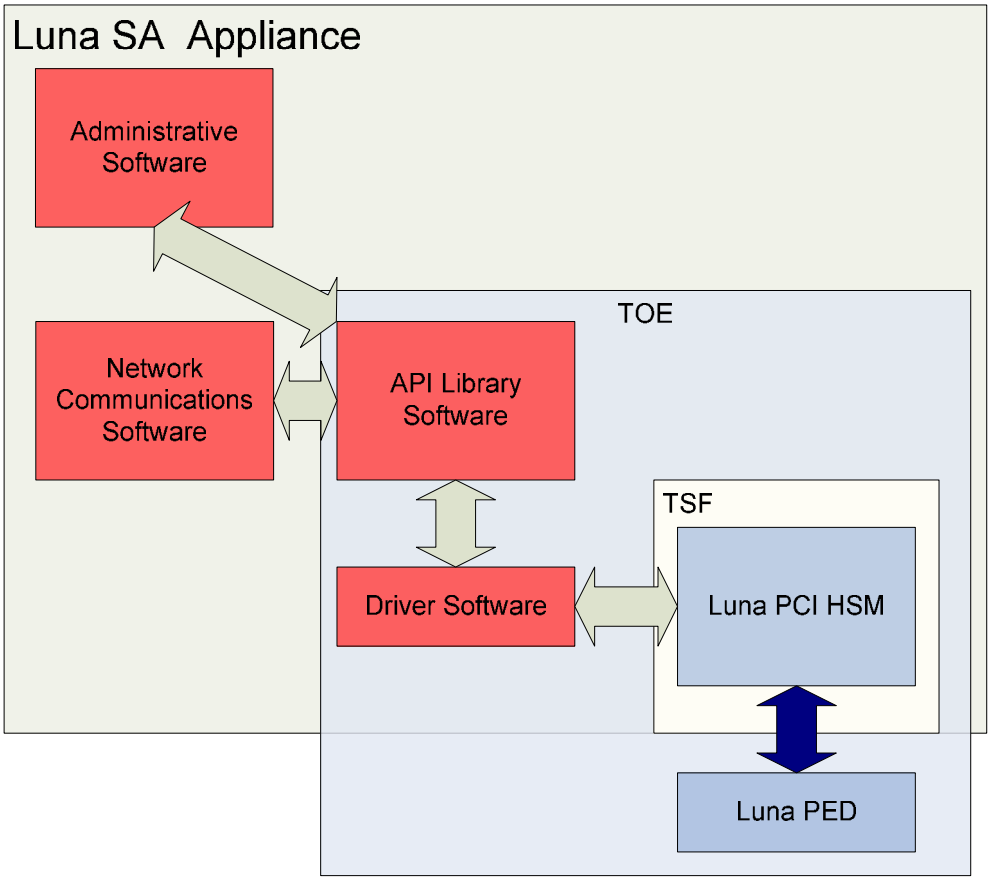

Figure 3 Illustration of the TOE and TSF in the Context of Luna SA

2.1. TOE Roles

The following authenticated roles are supported by the TOE:

- Security Officer (SO) – authorized to install and configure the TOE, set and maintain security policies, and create and delete users (Crypto Officer and Crypto User roles). The TOE can have only one SO.
- Crypto Officer – authorized to create, use, destroy and backup/restore cryptographic objects.
- Crypto User – authorized to use cryptographic objects (e.g., sign, encrypt/decrypt).

The Crypto Officer and Crypto User communicate with the Luna® PCI for cryptographic operations using the PKCS #11 API. The Security Officer uses a separate Command Line Interface (CLI), which is part of the interface software, to perform configuration, security policy settings and user creation/deletion. The CLI is also used by the Crypto Officer to perform backup and restoration of cryptographic objects.

The TOE allows for the creation of multiple users in the Crypto Officer and Crypto User roles. Each user is created within a cryptographically separated partition in the Luna® PCI cryptographic module and each partition must have one and only one user in the Crypto Officer role. A partition may also have one and only one user in the Crypto User role. Throughout the remainder of the ST, the term User will be used to refer to a partition user, in either the Crypto User or Crypto Officer role, when it is either not required or not appropriate to distinguish between the roles. The term user will be used to refer to a generic user, either unauthenticated or authenticated in any one of the 3 roles.

In **Table 2-1** the roles supported by the TOE are compared to the roles defined in the CMCSOB PP and PKCS#11.

Table 2-1 – CMCSOB PP to TOE Role Comparison

Function	PP Role	TOE Role	PKCS#11 Role
Initialisation, configuration	Crypto Officer	Security Officer	Security Officer
Key Management	Crypto Officer	Crypto Officer	User
Use	Crypto User	Crypto User	User

2.2. Cryptographic Services

The TOE provides the full range of cryptographic and key management functions. The major functions supported by the TOE are outlined below:

Random Number Generation

A trustworthy Random Number Generator is required to support secure generation of symmetric keys and asymmetric key pairs.

- FIPS 140-2 validated Deterministic Random Bit Generator (Pseudo-random Number Generator) seeded by internal Hardware Non-deterministic Random Bit Generator
- Based on ANSI X9.31, Appendix A section 2.4

Generate Public/Private Key Pairs

It is important that key pairs are properly generated in accordance with approved standards. The TOE provides key pair generation in accordance with the following standards.

- RSA 1024, 2048, 4096 bits key pairs in accordance with ANSI X9.31
- DSA 1024 bits key pairs in accordance with FIPS PUB 186-2
- ECDSA in accordance with FIPS PUB 186-2 and ANSI X9.62

Generate Secret (Symmetric) Keys

It is important that symmetric keys are properly generated in accordance with approved standards. The TOE provides key generation in accordance with the following standards.

- TDES 112, 168 bits in accordance with FIPS PUB 46-3 and ANSI X9.52
- AES 128, 192, 256 bits in accordance with FIPS PUB 197

Secure Key Material Storage and Access

Sensitive key values must be strongly protected and never be visible in plaintext form. The TOE ensures this in the following ways.

- Key material stored in hardware and strongly encrypted
- Access to private keys and symmetric keys is provided via key handles only

Compute Digital Signatures and Verify Digital Signatures

The TOE computes and verifies digital signatures in accordance with the following standards.

- RSA 1024 bits, 2048 bits, 4096 bits (PKCS #1 V1.5, PKCS #1 PSS, ANSI X9.31) with SHA-1
- RSA 1024 bits, 2048 bits, 4096 bits (PKCS #1 V1.5, PKCS #1 PSS) with SHA-256, 384, 512
- DSA 1024 bits (FIPS PUB 186-2) with SHA-1
- ECDSA (FIPS PUB 186-2 Appendix 6 recommended curves) with SHA-1

Encrypt / Decrypt Data

The TOE performs encryption and decryption operations on user data in accordance with the following standards.

- RSA 1024, 2048 and 4096 bits in accordance with PKCS #1 V1.5 and OAEP
- TDES (ECB and CBC mode) 112 and 168 bits in accordance with FIPS PUB 46-3
- AES (ECB and CBC mode) 128 and 256 bits in accordance with FIPS PUB 197

Import (Unwrap) Private Keys

The TOE can import private keys using an Unwrap operation in accordance with the following standard.

- RSA 1024, 2048 and 4096 bit private keys in PKCS #8 format with TDES and AES in CBC mode

Export (Wrap) and Import (Unwrap) Secret Keys

The TOE can export and import symmetric keys using Wrap and Unwrap operations in accordance with the following standards.

- TDES, AES with TDES and AES in ECB mode
- TDES, AES with RSA 1024, 2048 and 4096 bits in accordance with PKCS #1 V1.5

Handling of key material and the use of cryptographic functions must be done in accordance with the key management procedures and policies of the user organization.

2.3. Non-cryptographic Security Services

The TOE provides the following security services to support the protection of key material and cryptographic services:

- User authentication,
- Access control for the creation and destruction of keys,
- Access control for security administration functions,
- Access control for usage of keys with cryptographic functions,
- Self-test of the TOE.

2.4. Trusted Path – Luna® PED

User authentication data and other critical security parameters are protected through the use of a separate port and data path for their transfer, and by providing mechanisms to protect their confidentiality and integrity. Attached to this separate data port is the Luna® PIN Entry Device or Luna® PED.

The Luna® PED, with accompanying iKeys, is depicted in **Figure 4**. It houses a number of input/output interfaces that, in combination, provide a trusted path device for the communication of authentication data and critical security parameters to and from the Luna® PCI cryptographic module. The Luna® PED has a graphics display used to display status and prompt messages, and a challenge secret that is output by the cryptographic module at the time a partition is created [see sub-section 2.5]. It has a keypad used to enter simple responses (Yes/No/Enter) and to enter an optional PIN that is combined with the authentication data stored on a iKey as part of the authentication process. It has a USB receptacle for the input/output of data to the iKey and it has a serial communications cable that connects to the separate data port, which is wired directly to the cryptographic module. Because the PED has a direct serial communications interface to the cryptographic module, only local entry of iKey authentication data is possible.

The following types of iKey are used with the Luna® PED:

- Blue (SO) iKey – for the storage of SO authentication data,
- Black (User) iKey – for the storage of User authentication data,
- Red (Domain) iKey – for the storage of the cloning domain data, used to control the ability to clone from a cryptographic module to a backup token,
- Green (M of N) iKeys – used to store M of N secret shares, used for multi-person control of critical functions,

Any iKey, once data has been written to it, is an Identification and Authentication device and must be safeguarded accordingly by the administrative or operations staff responsible for the operation of the TOE within the customer's environment.

Figure 4 Luna® PED with PED Keys

2.5. User Authentication

The TOE requires that all users (SO, Crypto Officer and Crypto User roles) be authenticated by proving knowledge of a secret shared by the user and the cryptographic module.

The TOE generates the authentication secrets using its Pseudo-Random Number Generator (PRNG). For the SO, the authentication secret is a 48-byte random secret and it is generated at the time the cryptographic module is initialised. For Users, the authentication secrets consist of a 48-byte random secret and separate challenge secret(s); these are generated at the time the partition is created by the SO. The authentication secret(s) are provided to the operator via the Luna® PED and iKey, described in sub-section 2.4, and must be entered by the operator via the Luna® PED and via a logically separate trusted channel (in the case of the response based on the challenge secret) during the login process. Both the Crypto Officer and Crypto User use the same 48-byte random secret. If a Partition is created with Crypto Officer and Crypto User roles, a separate challenge secret is generated for each role.

SO authentication requires the transmission to the cryptographic module of the Blue (SO) iKey data combined with the optional PIN through the trusted path.

User authentication is a two-stage process. The first stage is termed “Activation” and is performed using the Luna® PED. Activation requires the transmission to the cryptographic module of the Black (User) iKey data combined with the optional PIN through the trusted path. Once Activation has been performed, the partition data is ready for use within the cryptographic module. Access to key material and cryptographic services, however, is not allowed until the second stage of authentication, equivalent to “User Login”, has been performed. This typically requires the input of a partition’s challenge secret as part of an application program’s login operation.

The authentication challenge secret (or secrets if the Crypto Officer and Crypto User roles are used) for the partition is generated by the cryptographic module as a 75-bit random value that is displayed as a 16-character string on the visual display of the trusted path device. The challenge secret is then provided, via a secure out-of-band means, to each external entity authorized to connect to the partition and is used by the external entity to form the response to a random one-time challenge from the cryptographic module. The encrypted one-time response is returned to the cryptographic module where it is verified to confirm the “User Login”.

2.6. Configurable Policy Settings

The Luna® PCI was designed with the flexibility needed to support a number of different product variants. The main method used to control the behaviour of different products is a fixed set of “capabilities” set at the factory. The settings that are possible to make for the TOE configuration are shown below. For each of the capabilities, a corresponding policy element exists. The TOE provides security management functions by giving the SO the ability to establish the policy that will govern the cryptographic module’s operation, according to the requirements of the customer organization, by enabling/disabling or refining the corresponding policy elements to equate to or to be more restrictive than the pre-assigned capabilities.

Policy set elements can only refine capability set elements to more restrictive values. Specifically, if a capability is set to allow, the corresponding policy element may be set to either enable or disable. However, if a capability is set to disallow, the corresponding policy element is set to disabled and is not SO-configurable. Thus, an SO cannot use policy configuration to lift a restriction set in a capability definition.

There are also several elements of the cryptographic module’s behaviour that are truly fixed for all product variants and, therefore, are never subject to configuration by the SO. These fixed elements are the following:

- Non-sensitive secret keys are not allowed.
- Non-sensitive private keys are not allowed.
- Non-private (Public) secret keys are not allowed.
- Non-private (Public) private keys are not allowed.
- Creation of secret keys and private keys through the PKCS #11 create object interface is not allowed. That is, the API cannot be used to create keys by passing in known plaintext values.

In the next two sub-sections, all capability elements described as “allow/disallow some functionality” are Boolean values where false (or zero) equates to disallow the functionality and true (or one) equates to allow the functionality. Except as noted, all Boolean capabilities are Allowed, thus leaving them configurable by the SO. The remainder of the elements are integer values with either the default value or the maximum in number of bits shown.

2.6.1. Cryptographic Module Capabilities

The following is the set of capabilities supported at the cryptographic module level:

- Allow/disallow non-FIPS algorithms available.
- Allow/disallow password authentication (disallowed in TOE configuration).
- Allow/disallow trusted path authentication (allowed and must be enabled in TOE configuration).
- Allow/disallow Remote PED Usage (disallowed in TOE configuration).
- Allow/disallow M of N.
- Allow/disallow cloning.

- Allow/disallow masking (disallowed in TOE configuration).
- Allow/disallow M of N auto-activation.
- Allow/disallow ECC mechanisms.
- Allow/disallow Remote Authentication.
- Allow/disallow SO reset of partition PIN.
- Allow/disallow network replication.
- Allow/disallow forcing change of User authentication data.
- Number of failed SO logins allowed before the HSM is zeroized (set to 3, non-configurable).

Note that the module policy "Allow/disallow Remote PED Usage" is disallowed with the TOE in the evaluated configuration. The TOE must be purchased in the "Local PED" configuration. Use of a Remote PED, which requires the module policy "Allow/disallow Remote PED Usage" to be changed to "Allow" means that the TOE can no longer be considered an evaluated configuration.

2.6.2. Partition Capabilities

The following is the set of capabilities supported at the partition level:

- Allow/disallow partition reset.
- Allow/disallow activation.
- Allow/disallow automatic activation.
- Allow/disallow High Availability.
- Allow/disallow multipurpose keys.
- Allow/disallow changing of certain key attributes once a key has been created.
- Allow/disallow operation without RSA blinding.
- Allow/disallow signing operations with non-local keys.
- Allow/disallow raw RSA operations.
- Allow/disallow private key wrapping (disallowed in TOE configuration).
- Allow/disallow private key unwrapping.
- Allow/disallow secret key wrapping
- Allow/disallow secret key unwrapping.
- Allow/disallow Level 3 operation without a challenge (disallowed in TOE configuration).
- Allow/disallow user key management capability. (Allowed in TOE configuration. This would be disabled by the SO at the policy level to prevent any key management activity in the partition, even by a user in the Crypto Officer role. This could be used, for example, at a CA once the root signing key pair has been generated and backed up, if appropriate, to lock down the partition for signing use only.)
- Allow/disallow incrementing of failed login attempt counter on failed challenge response validation.
- Allow/disallow RSA signing without confirmation.
- Allow/disallow RA type wrapping (disallowed in TOE configuration).
- Minimum/maximum password length (not applicable in TOE configuration).
- Level of storage space available for key storage (4 bits).
- Number of failed Partition User logins allowed before partition is locked out/cleared.(default is 10, SO can configure to be 3 <= N <= 10).

The following capabilities are only configurable if cloning is allowed and enabled at the cryptographic module level:

- Allow/disallow private key cloning (allowed in TOE configuration).
- Allow/disallow secret key cloning (allowed in TOE configuration).

The following capabilities are only configurable if masking is allowed and enabled at the cryptographic module level:

- Allow/disallow private key masking (disallowed in TOE configuration).
- Allow/disallow secret key masking (disallowed in TOE configuration).

2.7. TOE Usage

The TOE provides a physically and logically protected component for the performance of cryptographic functions such as key generation, key storage, encryption and decryption, digital signature and verification used by application systems that provide cryptographic support services such as a Certification Service Provider (CSP). It includes processors, read-only and random-access memory, and firmware packaged in a tamper-resistant form along with Cryptographic API software that resides on the host computer.

It is accessed directly (i.e., electrically) via either the PIN Entry Device (PED) serial interface or via the PCI bus interface. Logical access to key material and cryptographic services for users and user application software is provided indirectly through the Cryptographic API software on the host computer and on the network-attached Luna® SA appliance.

Before the TOE can be used to perform any cryptographic or key management functions, it must first be initialised. Initialisation causes the cryptographic module's contents (if any) to be erased and creates the SO for the cryptographic module. The SO must then set the configurable policies at the cryptographic module level and create at least one partition, with its corresponding user in the Crypto Officer role (creating a user in the Crypto User role is optional), to make the cryptographic module ready for use. There is no theoretical limit on the number of partitions that can be created by the SO. However, due to cryptographic module data storage constraints, a limit of 20 partitions per cryptographic module has been imposed. The SO may also be required to make policy settings at the partition level to conform to the organization's security requirements.

In operation, the TOE requires users in any of the 3 roles to be identified and authenticated before they are authorized to perform any cryptographic and/or key management operations. Authentication is performed using the Luna® PED or a combination of Luna® PED and a one-time challenge-response mechanism for the Crypto Officer and Crypto User roles.

In order to support requirements for strict separation of duties and/or multi-person control of critical security functions, the TOE also supports an optional M of N secret sharing mechanism in addition to the authentication data stored on the users' iKeys.

The TOE supports access by multiple users. Each user establishes one or more sessions with the cryptographic module, by which requests for services are transmitted to the cryptographic module and responses received. Session states are kept separate based on the user authentication state ensuring that sessions cannot be shared among users. Although there may be many users authenticated to the cryptographic module, there is effectively only one thread of execution within the module and, therefore, only one command being executed from request through to response at any given time.

2.8. Backup and Restoration

In order to support backup and transparent recovery of the cryptographic keys and supporting data stored within the Luna PCI cryptographic module, an optional backup and restoration capability can be provided. Each partition on the Luna PCI cryptographic module may have its cryptographic objects backed up to a Luna® Backup Token using the Luna® Key Cloning protocol. Conversely, the cryptographic objects stored on a Backup Token may be restored to a partition on a properly initialised Luna PCI cryptographic module.

2.9. Firmware Upgrade

The Luna PCI cryptographic module provides a capability to upgrade the module's firmware. Only the SO can perform a firmware upgrade. Each valid firmware upgrade package is digitally signed by SafeNet and encrypted. The customer must possess the correct authorization code in order to open the firmware upgrade package and the signature must be verified by the cryptographic module before the module will accept the upgrade.

2.10. User and Administrator Guidance Documentation

User and Administrator Guidance documentation is provided through the Luna on-line help system provided to the customer on CD-ROM as part of the delivered TOE.

2.11. Environment

The TOE is normally used as the cryptographic module for the Luna SA appliance. As such, it is delivered to the customer complete with the most important components of the environment. These environmental components provided with the Luna SA are the following:

- The Luna SA appliance platform running the CentOS 5.4 operating system.
- The Luna SA 4.5.1 software, which includes a Command Line Interface (CLI) that presents the visible user interface for administration of the TOE and the PKCS #11 Cryptographic API software, provided as a Linux shared-object library.
- A Luna Backup Token (PC Card) used to securely store cryptographic objects and TSF data needed for recovery from a failure.

The client PKCS #11 Cryptographic API software, provided as part of the TOE as a Windows DLL or Unix-type shared-object library depending on the host platform configuration, runs within the appliance environment and provides the programming interface to the host software application, which normally acts as the user of the TOE.

The TOE is normally operated in a physically secured environment by users who have been specifically authorized to do so by the owning organization. Because the TOE is typically used within a larger system, such as a Public Key Infrastructure (PKI), as part of a Certification Authority (CA) or Certification Service provider (CSP), the environment will often also include a variety of hardware, software, telecommunications and networking devices as well as uninterruptible power supplies and environmental controls.

Note also that the TOE does not, in itself, provide full security audit functionality. It does export the raw data (with identifying sequence numbers but without time stamps) needed to compile an audit record. If security audit is required for the system within which the TOE is operating, the host IT environment must provide the means of recording security relevant events, so as to assist an administrator in the detection of potential attacks or mis-configuration of the system, of which the TOE is a part, that could leave the host system and/or the TOE itself susceptible to attack, and also to hold users accountable for any actions they perform that are relevant to the security of the system.

3. TOE SECURITY ENVIRONMENT

3.1. Assets to protect

The primary assets that need to be protected by the TOE are the following:

Services ensured by the TOE:

R.Services: integrity and availability of the TOE services as well as protection against misuse is required.

TOE internal data:

R.User_Data: confidential user data (CSP-SCD, other user related private and secret keys (if any), etc.). Such data must be protected both in confidentiality and integrity. Data to be signed with CSP-SCD has to be protected in integrity.[2]

R.TOE_Data: TSF data (especially VAD and RAD) and other sensitive TOE data not related to a user or role (TOE configuration data, audit data) which have to be protected in confidentiality, integrity and availability.[3]

R.USERMGMT_DATA: non-confidential user / role related data (identifier, access control lists, role definitions, etc.). Those data has to be protected in integrity.

Data shared between the TOE and its environment:

R.Backup: backup data exported by the TOE to the TOE environment and restored in the TOE. This data needs to be protected in integrity and confidentiality by the TOE. Availability of this data has to be ensured in the TOE environment.

R.Exch_Data: data exchanged by the TOE through its interface (parameters for services that can be activated through the interface). They have to be protected in integrity. Some of those imported data shall also be protected in confidentiality (encipher keys, verification authentication data).

3.2. Assumptions

Some of the assumptions of this ST are either more specific than or in addition to those of the CMCSO PP. The following assumptions have been added to the ST for the reasons indicated.

1. A.Admin. Because of the overall complexity of the TOE, the personnel responsible for its administration (installation, configuration, audit review, etc.) must be adequately trained.
2. A.Controlled_Access. This ST assumes that physical and procedural access controls are in place and maps O.ENV_Protect_Access to this assumption. The CMCSO PP maps O.ENV_Protect_Access to a number of different threats to be countered by the TOE. Both approaches are equivalent. It is felt that the assumption of controlled access is more in keeping with the actual conditions present in the expected operating environment of the TOE.
3. A.Legitimate_FW_Update. The TOE allows the cryptographic module firmware to be updated, provided that the firmware update code is properly packaged and signed by the vendor. This assumption covers the proper packaging and signing of the firmware update, which must be done in the environment.
4. A.User_Management has been added and A.User_Authentication has been re-worded to more accurately reflect the actual conditions related to user authentication and the handling of user roles. These two assumptions are meant to capture the intent of A.User_Authentication and its associated Application Note in the PP.

[2] This is changed from the PP text to complete a sentence and improve readability.
[3] This is changed from the PP text to make it clear that it is TOE data that must be protected and not overall system data (which also includes data related to the environment). It also clarifies that it is sensitive data which must be protected.

A.Audit_Support *CSP audit review*

The CSP reviews the audit trail generated and exported by the TOE. The client application receives and stores the audit trail of the TOE for review by the System auditor of the CSP according to the audit procedure of the CSP.

A.Correct_DTBS *Correct DTBS Content Data*

The DTBS-representation submitted to the TOE is assumed to be correct. This requires that the DTBS (e.g. the certificate content data) has been generated and formatted correctly and maintains this correctness until it is passed to the TOE.[4]

A.Data_Store *Storage and Handling of TOE data*

The TOE environment ensures the confidentiality, integrity and availability of their security relevant data for TOE initialisation, start-up and operation if stored or handled outside the TOE. The TOE environment ensures the availability of the backup data. Examples of these data are verification authentication data, cryptographic key material and documentation of TOE configuration data.

A.Human_Interface *Interface with Human Users*

The client application will provide an appropriate interface and communication path between human users and the TOE because the TOE does not have a human interface for authentication and management services. The TOE environment transmits identification, authentication and management data of TOE users correctly and in a confidential way to the TOE.

A.User_Authentication *Authentication of Users*

The client application software is assumed to be operating as the TOE user on behalf of a human user and interacts directly, including authenticating, as the user of the TOE. Individual human users authorised to access the TOE cryptographic services may not be known to the TOE itself. The TOE environment performs identification and authentication for the individual users and allows successfully authenticated users to use the client application as their agent for the cryptographic services.[5]

A.Admin *Trustworthy TOE Administration*

When in operation, it is assumed that there will be a competent authority assigned to manage the TOE and the security of the information that it contains and who can be trusted not to deliberately abuse their privileges so as to undermine security.

A.Controlled_Access *Physical Security Controls*

When in operation and when stored as a backup, the TOE is assumed to be located within a controlled access facility providing physical security that is adequate to prevent physical access by unauthorized persons.

A.Legitimate_FW_Update *Legitimate Firmware Update Signed by the Vendor*

It is assumed that legitimate firmware update packages are digitally signed by the vendor using a private key whose use is restricted to this purpose and that the digital signature is verifiable by an instance of the TOE.

A.User_Management *User Management*

The TOE will not, in general, be aware of the identities of end-users authorised for the TOE services. It is assumed that the management of the individual user assignments for the 3 TOE roles is done in the environment in a trustworthy fashion according to a well-defined policy.

[4] The wording has been changed from the PP for readability. A portion of the assumption description in the PP has been omitted because it is not necessary. The omitted part describes the overall context in which the DTBS might be generated. While this is interesting, it is not essential to the assumption regarding DTBS correctness and could be interpreted as restricting the nature of the DTBS that may be signed by the TOE.

[5] The wording has been changed from the PP for clarity and readability.

3.3. Threats

Threat agents may include both unauthorized and authorized[6] users (persons or software entities) acting out of deliberate intent or through errors and omissions. Threat agents may also include malicious code of varying levels of sophistication, many of which are readily available on the Internet.

Relevant expertise required by threat agents may be in general semiconductor technology, software engineering, hacking techniques, and the resources may range from personal computers and peripherals to general-purpose test and measurement devices.

Motivation may include economic reward, a desire to damage an organization or the satisfaction and notoriety of defeating expert security.

Unauthorized users are assumed to be moderately motivated and to have a low to moderate level of relevant expertise and a low level of access to required resources. Attack potential for unauthorized users would be rated as Moderate.

Authorized users are assumed to be highly motivated (if they are acting deliberately) and to have at least a moderate level of relevant expertise and a high level of access to the required resources. Attack potential for authorized users would be rated as High.

The following threats have been added to those stated in the PP:

T.Exchange – This threat statement has been added because the TOE is often used in a role in which secure exchange of user data is important.

T.Key_Management – This threat statement has been added to emphasize the possibility of attacks through potential weaknesses in the TOE's key management functions.

T.PIN_Compromise – This threat statement has been added to emphasize the possibility of attacks to be mounted through the capture of user authentication data.

T.Unauth_Function – This threat statement has been added to address the potential to mount an attack on the TOE by causing unauthorized functions to be invoked. This could include, for example, discovery of a means to bypass the TSF or the use of a buffer overflow type of attack.

3.3.1. Threats Statements

T.Bad_FW_Load *Loading Malicious Firmware into the TOE*

An authorized user or an unauthorized user who has gained access to the TOE may modify the existing firmware and compromise the security functions of the TOE by loading unauthorized code.[7]

T.CSP-SCD_Derive *Deriving All or Parts of a Secret or Private Key*

An authorized user or an unauthorized user who has gained access to the TOE derives all or parts of a secret or private key, in particular the CSP-SCD, in any way (including the legitimate use of the TOE services).[8] This includes also any ability to derive all or part of the CSP-SCD using knowledge about the CSP-SCD generation and signing processes.

T.CSP-SCD_Disclose *Disclosing All or Part of a Secret or Private Key*

An authorized user or an unauthorized user who has gained access to the TOE discloses all or part of a secret or private key, in particular the CSP-SCD, over any physical or logical TOE interface.[7]

[6] Authorized user in this context means a person or software entity with organizational permission to access the TOE and its services.
[7] This threat is worded to make the general statement in the PP T.Bad_SW specific to the characteristics of the TOE.
[8] This sentence was re-worded to make the statement in terms of a threat agent acting through a vulnerability.

T.CSP-SCD_Distortion *Distortion of the CSP-SCD*

An authorized user or an unauthorized user who has gained access to the TOE signs the DTBS using a distorted CSP-SCD causing the signed data (e.g. qualified certificates or CRLs) to be invalid.[7] Although the use of a distorted CSP-SCD can be detected, the impacts for the organisation issuing the signed data using the CSP-SCD (e.g. qualified certificates) can be high. There is also the danger that by the use of a distorted CSP-SCD, parts of the original CSP-SCD can be derived.

T.Data_Manipul *Manipulating Data outside of the TOE*

User data that is transmitted to the TOE from the client application may be manipulated within the TOE environment before it is passed to the TOE. This may result in the effect that the TOE signs data without the approval of the user under whose control the data is submitted to the TOE. When performed within the client application such manipulations may not be detectable by the TOE itself and therefore this threat needs to be countered within the TOE environment.

T.Insecure_Init *Insecure Initialisation of the TOE*

Unauthorised CSP personnel or authorised CSP personnel without using adequate organisational controls may initialise the TOE with insecure system data, management data or user data.

An attacker may manipulate the backup data to initialise the TOE insecurely by the restore procedure.

T.Insecure_Oper *Insecure Operation of the TOE*

The TOE may be operated in an insecure way not detectable by the TOE itself. This includes the use and operation of the TOE within another environment than the intended one (e. g. the TOE may be connected to a hostile system).

T.Malfunction *Malfunction of TOE*

Internal malfunction of TOE functions may result in the modification of DTBS-representation, misuse of TOE services, disclosure or distortion of CSP-SCD or denial of service for authorized users. This includes the destruction of the TOE as well as hardware failures which prevent the TOE from performing its services. This includes also the destruction of the TOE by deliberate action or environmental failure. Technical failure may result in a insecure operational state violating the integrity and availability of the TOE services. The correct operation of the TOE also depends on the correct operation of critical hardware components. A failure of such a critical hardware component could result in the disclosure or distortion of the CSP-SCD, the modification of DTBS-representation or the ability to misuse services of the TOE. Critical components might be:

- the central processing unit
- a coprocessor for accelerating cryptographic operations
- a physical random number generator
- storage devices used to store the CSP-SCD or the DTBS-representation
- physical I/O device drivers

T.Management *Misuse of Management*

CSP personnel may misuse the TOE services to forge user data as CSP-SCD, user management data, system data or TSF data.

T.Misuse_Sign *Misuse of signature-creation function*

A user of the client application or of the TOE misuses the TOE service for signature-creation to sign with the CSP-SCD forged qualified certificates or forged certificate status information.

T.Phys_Manipul *Physical Manipulation of the TOE*

An unauthorized user who has gained access to the TOE may try to physically manipulate the TOE with the intent to discover all or part of a key stored or used within the TOE, to manipulate the DTBS within the TOE or to misuse services of the TOE.

T.Signature_Forgery *Forgery of digital signature*

An unauthorised user may exploit weaknesses in the cryptography and/or key management in the TOE in order to forge a digital signature in a way that is not detectable by the verifier of the signature.

T.Exchange *Compromise of Exchanged Data*

An unauthorized entity may gain access to sensitive user data or may cause undetected modifications of sensitive user data exchanged between the TOE and other IT entities.

T.Key_Management *Exploiting Weaknesses in Key Management*

An authorized user or an unauthorized user who has gained access to the TOE may compromise the security of key material in the cryptographic module by accidentally or deliberately exploiting weaknesses in key management, including key generation, storage and destruction.

T.PIN_Compromise *Compromise of Authentication data*

An unauthorized user may gain access to the authentication data of authorized users by intercepting the authentication data as it is entered or by guessing weak authentication data, and may impersonate an authorised user of the TOE.

T.Unauth_Function *Exploiting Unauthorized Functions*

An unauthorized person who has gained access to the TOE may reveal, discover or modify security data within the TOE by exploiting unauthorized functions of the TOE[9].

3.4. Organisational Security Policies

P.Algorithms *Use of Approved Algorithms and Algorithm Parameters*

Only algorithms and algorithm parameters (e. g. key length) defined as acceptable for being used for signature-creation by trustworthy systems shall be used to generate qualified certificates or to sign certificate status information. Where confidentiality protection is required such as for backup of CSP-SCD, only algorithms and algorithm parameters defined as acceptable for that purpose shall be used.

[9] An unauthorized function in this context is one which is not permitted in authorized versions of the TOE firmware, but which an unauthorized person is able to execute by causing malicious code to execute through some form of attack, such as a buffer overflow attack.

4. SECURITY OBJECTIVES

This section identifies and defines the security objectives for the TOE and its environment. Security objectives reflect the stated intent and counter the identified threats, as well as comply with the identified organisational security policies and assumptions.

Because the TOE has a broader range of functionality and potential applications than those envisaged by the CMCSOB PP, this ST has objectives that are more specific than or in addition to the objectives stated in the CMCSOB PP. The table below demonstrates that all of the objectives of the CMCSOB PP are also included in this ST and indicates those that are in addition to the objectives of the PP.

	CMCSOB PP Security Objectives								
ST Security Objectives	O.Audit_CM	O.CSP-SCD_Secure	O.Check_Operation	O.Control_Services	O.Detect_Attack	O.Error_Secure	O.Protect_Exported_Data	O.Sign_Secure	O.User_Authentication
O.Admin									
O.Approved_Algorithms		**X**					**X**	**X**	
O.Audit_CM	**X**								
O.Auth_Data_Protect									
O.Backup							**X**		
O.Check_Operation			**X**						
O.Control_Access				**X**					
O.Data_Exchange_Protect									
O.Detect_Attack					**X**				
O.Error_Secure						**X**			
O.Import_Code									
O.Key_Secure		**X**							
O.Multi-Person_Control									
O.Secure_Init									
O.Self_Protect									
O.Sign_Secure								**X**	
O.User_Authentication									**X**
O.User_Data_Protect									

The following objectives have been added or represent refinements of the PP objectives.

O.Admin – has been added to address the need for secure management of the TOE.

O.Approved_Algorithms – is a refinement of the PP objectives in that the statement requiring use of approved algorithms is found in three of the PP objectives and it has simply been pulled out to stand as its own objective statement.

O.Auth_Data_Protect – has been added to counter T.PIN_Compromise.

O.Backup and O.Data_Exchange_Protect – refine O.Protect_Exported_Data by setting separate objectives for backup and for the export of user symmetric keys and data for data exchange purposes.

O.Import_Code – has been added to counter T.Bad_FW_Load.

O.Key_Secure – expands the objective O.CSP-SCD_Secure to include all key material handled by the TOE.

O.Multi-Person_Control – has been added to assist in countering T.Key_Management and T.Management.

O.Secure_Init – has been added to assist in countering many of the threats by ensuring that the TOE is always in a secure state when it starts up.

O.Self_Protect – has been added to assist in countering many of the threats by ensuring that the TSF are protected from external influence and cannot be bypassed.

O.User_Data_Protect – has been added to address the protection of user data confidentiality and integrity by the TSF and to allow users to verify the authenticity of data.

Objectives from the PP are listed first, followed by the added or refined objectives.

4.1. Security Objectives for the TOE

O.Audit_CM *Generation and Export of Audit Data*

The TOE shall generate and export data needed to support audit record generation of the following events:

- TOE initialisation
- TOE start-up
- Generation of secret key and asymmetric key pair
- Destruction of key material
- Use of secret key and private key
- Unsuccessful authentication
- Modification of TOE management data
- Adding new users
- Deleting users
- Execution of the TSF self tests during initial start-up and at the request of the authorised user
- Unsuccessful self test operations
- ~~Reading and deleting audit trail records~~[10]
- Execution of the cloning functions for key backup and restore operations[11]

The audit data shall associate each auditable event with the identity of the user that caused the event. The integrity of the audit trail shall be ensured. The TOE shall export the audit data upon request the Auditor and the Crypto-officer. The TOE shall provide the management function for the audit to the Auditor only.

O.Check_Operation *Check for Correct Operation*

The TOE shall perform checks to verify that its components operate correctly and shall ensure the checks succeed before allowing any privileged or cryptographic operations by the TOE. This includes integrity checks of TOE software, firmware, internal TSF data or user data during installation, start-up and at the request of the authorised user.[12]

[10] Deleted from the list of audited events because it is the TSF itself that acts as the Auditor and deletes records when the circular buffer wraps around.
[11] This replaces the last 4 items in the PP list of audited events.
[12] The wording has been changed slightly from the PP for readability and to better reflect the TOE.

O.Control_Access *Access Control for Data and TOE Services*

The TOE shall control access to mediated commands, information, and services based on a user's identification and role, the requested command or service and the security attributes associated with the object to which access is requested. Roles and assignment of services to roles are predefined in the production phase.[13]

O.Detect_Attack *Tamper Resistance and Detection*

The TOE must be constructed in a manner that resists physical tampering and attempts at probing and shall detect tamper events and securely destroy all plaintext key material and other security critical data in this case.

O.Error_Secure *Secure State in Case an Error is Detected*

The TOE shall enter a secure state whenever it detects an error that would prevent its normal operation. The secure state shall prevent the loss of confidentiality of the key material and other security critical data.

O.Sign_Secure *Secure advanced signature-creation*

The TOE creates signatures such as the advanced signature in qualified certificates that do not reveal the CSP-SCD and cannot be forged without knowledge of the CSP-SCD.

O.User_Authentication *Authentication of Users interacting with the TOE*

The TOE shall be able to identify and authenticate the users acting with a defined role, before allowing any access to TOE protected assets and services. Identification and authentication shall be identity[14]-based.

O.Admin *TOE Administration*

The TOE must provide facilities to enable the Security Officer (SO) to effectively manage the TOE and its security functions, and must ensure that only authorized users are able to access such functionality.

O.Approved_Algorithms *Use of Approved Algorithms*

The TOE must implement approved cryptographic algorithms and algorithm parameters (e.g., key length) for encryption/decryption, authentication, and signature generation/verification.

O.Auth_Data_Protect *Protection of User Authentication data by the TOE*

The TOE must protect authentication data in a way that ensures that user authentication data cannot be easily guessed or captured.

O.Backup *Backup and Restore for the TOE*

The TOE shall protect the confidentiality of the backup data and detect loss of the integrity of the backup keys, other user data and TSF data needed to restore an operational state after failure.

O.Data_Exchange_Protect *Protection of Data Exported by the TOE*

The TOE shall provide integrity and confidentiality protection measures for all assets listed in the asset list requiring integrity or confidentiality protection when they are exported from or imported to the TOE.

O.Import_Code *Prevention of Unauthorised Code Loading*

The TOE must prevent executable code from being loaded on the TOE unless it is signed as per an authorized firmware update.

O.Key_Secure *Secure Key and Key Pair Generation and Management*

The confidentiality and integrity of cryptographic keys shall be ensured during their whole life-cycle. The TOE shall ensure secure key and key pair generation, use and management. This includes protection against disclosing completely or partly any cryptographic key through any physical or logical TOE interface.[15]

[13] This is a generalized form of O.Control_Services that addresses all aspects of access control.
[14] User-based has been changed to identity-based for accuracy (a user can be authenticated by either role or identity) and to ensure common usage with the FIPS 140-2 Level 3 Security Policy requirements.
[15] This is a generalized form of O.CSP-SCD_Secure that addresses the security of all key material, not only the CSP-SCD.

O.Multi-Person_Control *Multi-person Control of Sensitive Functions*

The TOE must provide a capability for multi-person control of sensitive functions.

O.Secure_Init *Secure Initialisation of TOE*

The TOE must assume its initial secure state immediately upon power-up, reset, or after other restart conditions.

O.Self_Protect *TOE Logical Self-Protection*

The TOE must protect itself against attempts to logically subvert or bypass the TOE security functions.

O.User_Data_Protect *Protection of User Data by the TOE*

The TOE shall protect the confidentiality and integrity of user data stored within the TOE and shall provide the means for the user to verify the authenticity of stored data.

4.2. Security Objectives for the IT Environment

O.ENV_Application *Security in the Client Application*

The applications which use the TOE shall perform the necessary security checks on the data passed to the TOE. The applications shall also perform the required user authentication and access control functions that cannot be performed within the TOE. Security controls in the TOE environment shall also prevent unauthorised manipulation of data submitted to the TOE.

O.ENV_Audit *Audit Trail Generation*

The environment ensures the availability of the generated and exported by the TOE audit trails and provides a review of the audit trail recorded by the TOE.

O.ENV_Backup *Backup and Restore Protection in the Environment*

The IT environment shall provide a means to protect the confidentiality of the backup data and detect loss of the integrity of the backup keys, other user data and TSF data needed to restore an operational state after failure when it is transmitted and stored in the TOE environment.

O.ENV_Human_Interface *Reliable Human Interface*

If the client application provides a human interface and a communication path between human users and the TOE, the client application will ensure the confidentiality and integrity of the data transferred between the TOE and the human user.

O.ENV_Signed_FW_Update *Firmware Updates Signed by the Vendor*

Procedures shall exist to ensure that legitimate firmware update packages are digitally signed by the vendor using a private key whose use is restricted to this purpose and that the digital signature is verifiable by an instance of the TOE.

4.3. Security Objectives for the non-IT Environment

O.ENV_AuthData *Personal Protection of Authentication Data*

Those responsible for the TOE must ensure that the authentication data for each user account for the TOE is held securely and not disclosed to persons not authorized to use that account.

O.ENV_Outage_Protection *Protection From Unplanned Outages*

Those responsible for the host IT environment must ensure that the power supplied to the TOE is adequately protected against unexpected interruptions and the effects of surges and voltage fluctuations outside the normal operating range of the device and that the TOE is operated in an environment that is provided adequate protection against disasters such as fire and flood.

O.ENV_Personnel *Reliable Personnel*

The personnel using the TOE services shall be aware of civil, financial and legal responsibilities, as well as the obligations they have to face, depending on their role. The personnel shall be trained on correct usage of the TOE and have a level of competence sufficient to ensure the correct management and operation of the TOE.

O.ENV_Protect_Access *Prevention of Unauthorised Physical Access*

The TOE shall be protected by physical, logical and organisational protection measures to restrict access to the TOE and its IT environment to authorised persons only, in order to prevent any TOE theft, modification or disclosure of protected assets.

O.ENV_Recovery *Secure Recovery in Case of Major Failure*

Recovery plans and procedures shall exist that allow a secure and timely recovery in the case of a major problem with the TOE. These procedures shall ensure that the confidentiality and integrity of TOE assets are maintained during recovery and that the recovery does not result in a situation that allows personnel to extend the TOE services they are allowed to use.

O.ENV_Secure_Init *Secure Initialisation Procedures*

Procedures and controls in the TOE environment shall be defined and applied that allow to securely set-up and initialise the TOE for the generation of signatures for qualified certificates or certificate status information. This includes the secure key generation / key import as well as the initial configuration of other TSF data such as roles, users and user authentication information.

O.ENV_Secure_Oper *Secure Operating Procedures*

Procedures and controls in the TOE environment shall be defined that allow operating the TOE within a CA system in compliance with the requirements of the EU directive and the Policy for certification authorities issuing qualified certificates.

4.4. Mapping of Objectives

The correspondence or mapping showing the necessity of the security objectives to counter the threats and meet stated assumptions is shown in Table 8-1. Tables relating assumptions and threats to objectives as well as the rationale for the sufficiency of the objectives is presented in Table 8-2 and Table 8-3.

5. IT SECURITY REQUIREMENTS

This chapter gives the security functional requirements (SFR) and the security assurance requirements (SAR) for the TOE and the IT environment.

Security functional requirements components given in section 5.1 TOE Security Functional Requirements match those of the CMCSOB PP and are drawn from the Common Criteria (ISO 15408) part 2 [2]. Some security functional requirements represent extensions to [2], with the rationale for these given with the explicit requirements statement. Operations for assignment, selection and refinement have been made.

Security functional requirements components given in section 5.2 are additional to those of the CMCSOB PP and are drawn from the Common Criteria (ISO 15408) part 2 [2].

Section 5.3 Requirements for IT Environment identifies the IT security requirements that are to be met by the IT environment of the TOE.

Section 5.4 Non-IT Requirements specifies the non-IT requirements to be satisfied by the environment.

TOE Security Assurance Requirements, section 5.5, are drawn from the security assurance components from Common Criteria part 3 [3].

The following non Part 2 Security Functional Requirements are included to meet specific requirements of the TOE:

- FCS_RND.1 Quality metrics for random numbers
- FDP_BKP.1 Backup and recovery

The following convention is used to indicate operations that have been performed on the CC functional components:

- Assignment is indicated by **bold** lettering.
- Selection is indicated by <u>underlining</u> the selection(s).
- Refinement is indicated by *italic* lettering.
- Iterations are indicated by supplementary bracketed information with the functional component, such as **FIA_AFL.1.1 (SO) and FIA_AFL.1.1 (User).**
- Deletion of PP wording is indicated by ~~strikeout~~ of the words in question. Rationale for such deletion is provided in the footnotes.

5.1. TOE Security Functional Requirements

5.1.1. Security audit (FAU)

5.1.1.1. FAU_GEN.1 Audit data generation

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

a) Start-up and shutdown of the audit functions;

b) All auditable events for the <u>not specified</u> level of audit; and

c) **The following specifically auditable events:**

(1) **Initialisation of the TOE,**

(2) **Start-up after power up,**

(3) ~~**Shutdown of the TOE**~~[16]**,**

(4) Cryptographic key generation (FCS_CKM.1),

(5) Cryptographic key distribution (FCS_CKM.2): negotiation of back-up key(s) as part of the cloning protocol,

(6) Cryptographic key destruction (FCS_CKM.4),

(7) Cryptographic operation (FCS_COP.1 (Sign),

(8) Authentication failure handling (FIA_AFL.1): reaching the threshold for the unsuccessful authentication attempts,

(9) Timing of authentication (FIA_UAU.1): all unsuccessful use of the authentication mechanism,

(10) Management of security attributes (FMT_MSA.1): all modifications of the values of security attributes,

(11) Static attribute initialisation (FMT_MSA.3): modifications of the default setting of permissive or restrictive rules, all modifications of the initial values of security attributes,

(12) Management of TSF data (FMT_MTD.1): modification to the threshold for unsuccessful User authentication attempts,

~~**(13) Management of TSF data (FMT_MTD.1/ACCESS_CONTROL): All modifications to the values of TSF data**~~[17]

~~**(14) Management of TSF data (FMT_MTD.1/AUDIT: Export of audit data, Clear of audit data,**~~[18]

(15) Abstract machine testing (FPT_AMT.1): results of the tests of the underlying abstract machine,

~~**(16) Failure with preservation of secure state (FPT_FLS.1): Failure detection of the TSF and secure state,**~~[19]

(17) Inter-TSF detection of modification (FPT_ITI.1): The detection of modification of imported TSF backup data

(18) Notification of physical attack (FPT_PHP.2): Detection of intrusion

(19) TSF testing (FPT_TST.1): Execution of the TSF self tests and the results of the tests,

(20) Backup and recovery (FDP_BKP.1): Use of the backup and recovery functions,

Refined by adding:

The auditable event Start-up and shutdown of the audit functions is not applicable as these are always present.

Audit data generated by the TOE is in raw form and requires interpretation by an application in the TOE environment before it can be included in a security audit trail.

Unless invoked by an authorised user, results of abstract machine tests and self-tests are only reported in the case of a failure.

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

[16] It is not possible to audit shutdown because there is no shutdown command. Shutdown is done by simply removing power and there is no opportunity to write an audit record in this case.
[17] FMT_MTD.1 (Access Control) does not apply to the TOE.
[18] These items are not applicable in the TOE.
[19] There is no specific TOE event corresponding to FPT_FLS.1.

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event. *Date and time of the event shall be given by the sequence data correlated to time of export of the audit data to the TOE environment*; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, **identity of the user and sequence data.**

Refined by adding:

The audit data for the Crypto Officer and Crypto User roles may only be sufficient to identify the client application. Further refinement of audit data might be provided by audit functions in the TOE environment distinguishing between end-users using the services of the client application.

5.1.1.2. FAU_GEN.2 User identity association

FAU_GEN.2.1 The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.1.1.3. FAU_STG.2 (TOE) Guarantees of audit data availability

FAU_STG.2.1 (TOE) The TSF shall protect the stored audit records from unauthorised deletion.

FAU_STG.2.2 (TOE) The TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail.

FAU_STG.2.3 (TOE) The TSF shall ensure that **60 kB of** audit records will be maintained when the following conditions occur: audit storage exhaustion.

Application note:

The TSF writes audit data to a circular buffer and overwrites the first record in the audit trail data after reaching the end of the buffer. Provided that the audit data is read frequently enough based on the typical operation rate of the customer application, there is effectively no reason for the audit data storage to be exhausted.

5.1.2. Cryptographic support (FCS)

5.1.2.1. FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with *the* specified cryptographic key generation *algorithms* **listed below** and specified cryptographic key sizes **specified for each algorithm** that meet the following **standards noted for each algorithm**:

(1) RSA 1024, 2048, 4096 bits key pairs in accordance with ANSI X9.31.
(2) TDES 112, 168 bits in accordance with FIPS PUB 46-3 and ANSI X9.52.
(3) AES 128, 192, 256 bits in accordance with FIPS PUB 197.
(4) DSA 1024 bits key pairs in accordance with FIPS PUB 186-2.
(5) ECDSA in accordance with FIPS PUB 186-2 and ANSI X9.62.

5.1.2.2. FCS_CKM.2 (BACKUP) Cryptographic key distribution

FCS_CKM.2.1 (BACKUP) The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **key entry** that meets the following: **TDES 168 bits**.

Refined by adding:

All encrypted secret or private keys entered into the TOE shall be encrypted using a cryptographic algorithm from the list of approved algorithms and parameters [5]. The key entry shall be performed using either manual or electronic methods.

~~*Secret and private keys established using manual methods shall be entered either*~~

~~*(1) in encrypted form or*~~

~~*(2) using split knowledge procedures.*~~

~~*Manually-entered keys shall be verified during entry into the TOE for accuracy.*~~

Secret and private keys established using electronic methods shall be entered in encrypted form.

~~*If split knowledge procedures are used:*~~

~~*(1) The TOE shall separately authenticate the crypto-officer entering each key component.*~~

~~*(2) At least two key components shall be required to reconstruct the original cryptographic key.*~~

Application note:

The Luna® Key Cloning Protocol is used to negotiate a session key to encrypt the data transferred between the TSF and the backup token. The protocol uses cryptographic techniques to provide mutual authentication, proof of origin, integrity and confidentiality of the objects being transferred from source to target token within a domain. The key management scheme used within the cloning protocol also protects against replay attacks and minimizes the impact of possible key compromise by ensuring that a unique AES key is used for each cloning operation.

5.1.2.3. 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 **logical or physical (by overwriting) deletion of the memory space** that meets the following: **FIPS 140-2 Level 3.**

Application note:

The TSF will destroy the CSP-SCD and all other plaintext secret or private keys, if the TSF required by FPT_PHP.2 detects physical tampering.

5.1.2.4. FCS_COP.1 (SIGN) Cryptographic operation - Digital signature

FCS_COP.1.1 (SIGN) The TSF shall perform **digital signature generation and verification** in accordance with *the* specified cryptographic *algorithm*s **listed below** and cryptographic key sizes **specified for each algorithm** that meet the following: **standards noted for each algorithm**.

(1) RSA 1024 bits, 2048 bits, 4096 bits with SHA-1, SHA-256, SHA-384, SHA-512 (PKCS #1 V1.5, PKCS #1 PSS),
(2) RSA 1024 bits, 2048 bits, 4096 bits with SHA-1 (FIPS PUB 186-2/ANSI X9.31),
(3) Signature generation/verification DSA 1024 bits with SHA-1 (FIPS PUB 186-2),
(4) Signature generation/verification ECDSA with SHA-1 (FIPS PUB 186-2 Appendix 6 recommended curves).

5.1.2.5. FCS_COP.1 (BACKUP_ENC) Cryptographic operation

FCS_COP.1.1 (BACKUP_ENC) The TSF shall perform **encryption and decryption** in accordance with a specified cryptographic algorithm **TDES** and cryptographic key sizes **168 bits** that meet the following: **FIPS PUB 46-3**.

5.1.2.6. FCS_COP.1 (BACKUP_INT) Cryptographic operation

FCS_COP.1.1 (BACKUP_INT) The TSF shall perform **calculation and verification of cryptographic checksums** in accordance with a specified cryptographic algorithm **SHA-1** and cryptographic key sizes **N/A** that meet the following: **FIPS PUB 180-2**.

Refined by adding:

~~The cryptographic checksum for backup data shall use a backup key and shall be based on symmetric cryptographic algorithms (e.g. keyed hash) or asymmetric cryptographic algorithms (e.g. digital signatures).~~

Application Note:

The SHA-1 checksum is calculated by the TSF and by the equivalent function within the backup token. It can be compared before and after backup and before and after recovery to validate integrity. Because it is calculated within the TSF and backup token at the time it is requested, there is no chance of substituting an illegitimate key value and corresponding digest value. A keyed hash or digital signature is, therefore, not required.

FCS_RND Generation of random numbers

Family behaviour

This family defines quality metrics for generating random numbers intended for cryptographic purposes.

Component levelling

FCS_RND.1 The generation of random numbers by the TOE requires the random numbers to meet the defined quality metrics.

Management

There are no applicable management functions.

Audit

There are no auditable events applicable to FCS_RND generation of random numbers.

5.1.2.7. FCS_RND.1 Quality metrics for random numbers

Hierarchical to: no other components.

FCS_RND.1.1 The TOE shall provide a mechanism for generating random numbers that meet **the requirements of ETSI SR 002 176 V1.1.1, dated 2003-03.**

FCS_RND.1.2 The TOE shall be able to enforce the use of TOE-generated random numbers for **FCS_CKM.1, authentication data generation and random number generation.**

Dependencies: FPT_TST.1 TSF testing.

Rationale:

High quality random numbers are important to key and key pair generation. This component allows for the specification of a particular quality metric to be met by the TOE and enforces the use of the specified random number generator.

5.1.3. User data protection (FDP)

5.1.3.1. FDP_ACC.1 (CRYPTO) Subset access control

FDP_ACC.1.1 (CRYPTO) The TSF shall enforce the **Crypto-SFP** on **User; CSP-SCD, CSP-SVD, DTBS representation; generate CSP-SCD/CSP-SVD pair (FCS_CKM.1), destruction of CSP-SCD and CSP-SVD (FCS_CKM.4); sign DTBS representation (FCS_COP.1(SIGN)).**

5.1.3.2. FDP_ACC.1 (AUDIT) Subset access control

FDP_ACC.1.1 (AUDIT) The TSF shall enforce the **Audit-SFP** on **User; Audit data; export and delete**.

5.1.3.3. FDP_ACC.1 (BACKUP) Subset access control

FDP_ACC.1.1 (BACKUP) The TSF shall enforce the **Backup SFP** on **User; CSP-SCD, backup key(s), backup data; backup (FDP_BKP.1), restore (FDP_BKP.1), backup key entry (FCS_CKM.2)**.

5.1.3.4. FDP_ACF.1 (CRYPTO) Security attribute based access control

FDP_ACF.1.1 (CRYPTO) The TSF shall enforce the **CRYPTO SFP** to objects based on **Identity and Role**.

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

1. **User with security attribute Role Crypto Officer is allowed to generate (FCS_CKM.1) the objects CSP-SCD and CSP-SVD under dual person control.**
2. **User with security attribute Role Crypto Officer is allowed to destruct (FCS_CKM.4) the objects CSP-SCD and CSP-SVD.**
3. **User with security attribute Role Crypto Officer is allowed to export CSP-SVD.**
4. **User with security attribute Role Crypto User is allowed to create signature of the DTBS-representation with CSP-SCD (FCS_COP.1/SIGN).**
5. **User with security attribute Role Crypto User is allowed to export CSP-SVD.**

FDP_ACF.1.3 (CRYPTO) The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **none**.

FDP_ACF.1.4 (CRYPTO) The TSF shall explicitly deny access of subjects to objects based on the **following rules: User with security attribute Role Crypto User is not allowed to:**

(a) generate (FCS_CKM.1) the objects CSP-SCD and CSP-SVD,

(b) destroy (FCS_CKM.4) the objects CSP-SCD and CSP-SVD.

Application note:

The dual person control requires two users to be authenticated – one with the role Crypto Officer and one with the role Security Officer. The Security Officer must ensure that the partition policy for Key Management Operations is set properly to enforce the dual control.

5.1.3.5. FDP_ACF.1 (AUDIT) Security attribute based access control

FDP_ACF.1.1 (AUDIT) The TSF shall enforce the **AUDIT SFP** to objects based on **Role**.

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

1. **Users with security attribute Role Auditor are allowed**
    a. **to export Audit data,**
    b. **to clear Audit data.**
2. **Users with security attribute Role Crypto Officer are allowed to export Audit data.**

FDP_ACF.1.3 (AUDIT) The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **none.**

FDP_ACF.1.4 (AUDIT) The TSF shall explicitly deny access of subjects to objects based on the **following rules:**

1. **Users with security attribute Role Crypto Officer are not allowed to delete Audit data**
2. **Users with security attribute Role Crypto User are not allowed to export or to delete Audit data.**

Application Note:

In the TOE, the Auditor role is held by the TSF. Users are only provided read access to audit data. This is obtained via an administrative interface command.

5.1.3.6. FDP_ACF.1 (BACKUP) Security attribute based access control

FDP_ACF.1.1 (BACKUP) The TSF shall enforce the **BACKUP SFP** to objects based on **Identity and Role**.

FDP_ACF.1.2 (BACKUP) The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **User with security attribute Role Crypto Officer is allowed under dual person control:**

1. **to backup CSP-SCD and CSP-SVD (FDP_BKP.1),**
2. **to restore CSP-SCD and CSP-SVD (FDP_BKP.1),**
3. ~~**to enter backup keys (FCS_CKM.2)**~~

Application Note:

The CKM.2 key entry requirement is met in the TOE by key negotiation between the TSF and the backup token. Therefore, there is no user involvement in key entry.

FDP_ACF.1.3 (BACKUP) The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **User with security attribute Role Crypto Officer is allowed under dual person control to backup all partition objects.**

FDP_ACF.1.4 (BACKUP) The TSF shall explicitly deny access of subjects to objects based on the **User with security attribute Role Crypto User is not allowed:**

1. **to backup CSP-SCD (FDP_BKP.1),**
2. **to restore CSP-SCD (FDP_BKP.1),**
3. **to enter a backup key (FCS_CKM.2).**

Application note:

If the TSF implementing FDP_BKP.1 does not support separate backup for CSP-SCD and for other backup data the additional rules in FDP_ACF.1.3 may allow the Crypto Officer to backup and to restore all backup data.

Backup and recovery (FDP_BKP)
Family behaviour

This family defines export and import of the backup data. The TOE ensures the confidentiality of the backup data and detects loss of the integrity of the backup data. The availability of the backup data will be ensured by the TOE environment.

Management: FDP_BKP.1
There are no management activities foreseen.

Audit: FDP_BKP.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
a) Use of the backup function,
b) Use of the recovery function,
c) Unsuccessful recovery because of detection of modification of the backup data.

5.1.3.7. FDP_BKP.1 Backup and recovery

FDP_BKP.1.1 The TSF shall be capable of invoking the backup function on demand.

FDP_BKP.1.2 The data stored in the backup shall be sufficient to recreate the state of the TOE at the time the backup was created using only:

1. a copy of the same version of the TOE as was used to create the backup data;
2. a stored copy of the backup data;
3. the cryptographic key(s) needed to decrypt the CSP-SCD and any other encrypted critical security parameters;
4. ~~the cryptographic key(s) needed to verify the cryptographic checksum of the backup data.~~

FDP_BKP.1.3 The TSF shall include a recovery function that is able to restore the state of the TOE from a backup.

FDP_BKP.1.4 The CSP-SCD, other critical security parameters and other confidential information shall be exported in encrypted form only.

FDP_BKP.1.5 The backup data shall be checked for modification through the use of cryptographic checksums. Modified backup data shall not be used for recovery.

Dependencies: [FCS_CKM.1 Cryptographic key generation or FCS_CKM.2 Cryptographic key distribution or FDP_ITC.1 Import of user data without security attributes] FCS_COP.1 Cryptographic operation

Rationale:

The HSM supports backup of CSP-SCD, other user data and TSF data to restore the operational state of the same HSM or for a new HSM in the event of a system failure or other serious error. The export, import and protection of the backup data are combined in a specific way. The HSM ensures the confidentiality of the backup data and detects loss of the integrity of the backup data. The availability of the backup data will be ensured by the TOE environment.

This component is necessary to specify a unique requirement of certificate issuing and management components that is not addressed by the Common Criteria. The specific requirements address the protection of CSP-SCD, other cryptographic keys and TSF data for backup and recovery.

Application Note**:**

The SHA-1 checksum is calculated by the TSF and by the equivalent function within the backup token. It can be compared before and after backup and before and after recovery to validate integrity. Because it is done within the TSF and backup token, there is no chance of substituting an illegitimate key value and corresponding digest value. A keyed hash or digital signature is, therefore, not required.

5.1.3.8. FDP_ETC.1 Export of user data without security attributes

FDP_ETC.1.1 The TSF shall enforce the **Crypto SFP, Token Access Control (TAC) SFP** when exporting user data, controlled under the SFPs, outside of the TSC.

FDP_ETC.1.2 The TSF shall export the user data without the user data's associated security attributes.

Application Note:

All user data whose CKA_SENSITIVE attribute is set is exported in encrypted form.

5.1.3.9. FDP_IFC.1 (BACKUP) Subset information flow control

FDP_IFC.1.1 (BACKUP) The TSF shall enforce the **Side-channel of backup-functions SFP** on **Anybody; Information about CSP-SCD; backup (FDP_BKP.1, FCS_COP.1/BACKUP_ENC, FCS_COP.1/BACKUP_INT), restore (FDP_BKP.1, FCS_COP.1/BACKUP_ENC, FCS_COP.1/BACKUP_INT), key entry (FCS_CKM.2).**

5.1.3.10. FDP_IFC.1 (CRYPTO) Subset information flow control

FDP_IFC.1.1 (CRYPTO) The TSF shall enforce the **Side-channels of Crypto-functions SFP** on **Anybody; Information about CSP-SCD; generation of CSP-SCD/SVD pair (FCS_CKM.1), destruction of CSP-SCD (FCS_CKM.4), signing DTBS representation (FCS_COP.1/SIGN).**

5.1.3.11. FDP_IFF.4 (BACKUP) Partial elimination of illicit information flows

FDP_IFF.4.1 (BACKUP) The TSF shall enforce the **Side-channel of backup-functions SFP** to limit the capacity of **covert channels information flow of:**

1. **the backup function including encryption of the backup data (FDP_BKP.1),**
2. **the backup key(s) entry (FCS_CKM.2),**
3. **the encryption and decryption of the backup data (FCS_COP.1/BACKUP_ENC)**

through physical behaviour of the TOE interfaces and emanation [assignment: **none**] **compromising information about the CSP-SCD** to a **maximum 0 bits/sec for covert channels and resistance to an attacker with attack potential 'high' for side channels.**

FDP_IFF.4.2 (BACKUP) The TSF shall prevent the **following types of side-channels information flow within the backup data (FDP_BKP.1) about the CSP-SCD.**

Application note:

The TOE shall prevent side-channel attacks against the CSP-SCD and other secret data where the attack is based on external observable physical phenomena of the TOE as mentioned in the application note to FDP_IFF.4/Crypto. The maximum capacity of the side channels shall be defined by the ST allowing the SCP to prevent any remaining side channels by appropriate security measures in the TOE environment. The TOE shall prevent side-channel attacks against the CSP-SCD through the intended output data of the TOE e.g. the backup data encrypted with an initial vector containing information about the used backup key.

5.1.3.12. FDP_IFF.4 (CRYPTO) Partial elimination of illicit information flows

FDP_IFF.4.1 (CRYPTO) The TSF shall enforce the **Side-channels of Crypto-functions SFP** to limit the capacity of **side-channels information flow of (1) the CSP-SCD/SVD generation (FCS_CKM.1), (2) the signature-creation (FCS_COP.1/SIGN), through physical behaviour of the TOE interfaces and emanation** [assignment: **none**] **compromising information about the CSP-SCD** to a **maximum 0 bits/sec for covert channels and resistance to an attacker with attack potential 'high' for side channels.**

FDP_IFF.4.2 (CRYPTO) The TSF shall prevent **side-channels information flow within the data exported**

(1) by the TSF CSP-SCD / SVD pair generation (FCS-CKM.1),

(2) by the TSF signature-creation function (FCS-COP.1/SIGN) about the CSP-SCD.

Application note:

The TSF requires the TOE to prevent side-channel attacks against the CSP-SCD and other secret data where the attack is based on external observable physical phenomena of the TOE. The set of measurable physical phenomena is influenced by the technology employed to implement the TOE. Examples of measurable phenomena are variations in the timing of transitions of internal states, the power consumption and the electromagnetic radiation. Such phenomena may be caused by normal internal operation of the TOE or may be forced by an attacker who varies the physical environment under which the TOE operates (e. g. power supply, temperature, radio emission or emission of light). Due to the heterogeneous nature of the technologies that may cause such emanations, evaluation is assumed against state-of-the-art attacks applicable to the technologies employed by the TOE. Examples of such attacks are, but are not limited to, evaluation of the TOE's electromagnetic radiation, simple power analysis (SPA), differential power analysis (DPA), timing attacks, etc. The maximum capacity of the side channels should be defined by the ST allowing the CSP to prevent any remaining side channels by appropriate security measures in the TOE environment. The TSF requires the TOE to prevent side-channel attacks against the CSP-SCD through the intended output data of the TOE e.g. the random padding bits in the signature may contain information about the CSP-SCD if both are generated by the same pseudo-random number generator.

5.1.3.13. FDP_RIP.1 Subset residual information protection

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the de-allocation of the resource from the following objects: **RAD, private keys, in particular CSP-SCD, and secret keys**.

5.1.3.14. FDP_SDI.2 Stored data integrity monitoring and action

FDP_SDI.2.1 The TSF shall monitor user data stored within the TSC for **integrity errors** on all objects, based on the following attributes: **error detecting code**.

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall **enter the secure blocking state[20].**

Refined by adding:

The TSF are not required to monitor the DTBS representation for integrity errors.

Application Note:

The error detecting code to check the integrity of a private key object is generated before output of a digital signature by verifying the digital signature created with the private key using the corresponding public key object for signature verification.

[20] In this context, secure blocking state means a state where the only data returned to the user is an error code and the TOE does not continue to produce a signature value.

5.1.4. Identification and authentication (FIA)

5.1.4.1. FIA_AFL.1 (SO) Authentication failure handling

FIA_AFL.1.1 (SO) The TSF shall detect when **three (3)** unsuccessful authentication attempts occur related to **Security Officer authentication**.

FIA_AFL.1.2 (SO) When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall **change the state of the TOE to require re-initialization**.

5.1.4.2. FIA_AFL.1 (User) Authentication failure handling

FIA_AFL.1.1 (User) The TSF shall detect when **an *SO* configurable positive integer within the range of three (3) to ten (10)** unsuccessful authentication attempts occur related to **User authentication**.

FIA_AFL.1.2 (User) When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall **block the identity for authentication.**

Application Note:

The TOE blocks the identity for authentication by terminating the session establishment and, according to the SO configurable policy:

- removing the User and clearing the User's memory space and permanent storage, or
- disabling the User account by setting the User locked flag in the User's attributes (FIA_ATD.1).

5.1.4.3. FIA_ATD.1 User attribute definition

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users:
User ID number
User checkword (RAD)
User role
User failed login count
User "locked" flag.

Application Note:

User role is derived from the type of PED Key used – Blue Key for SO and Black Key for User, plus the type of Challenge-response authentication – either Crypto Officer or Crypto User. The Crypto Officer and Crypto User each have a different challenge secret, which is stored encrypted by the User Security Key (retrieved from the encrypted checkword).

5.1.4.4. FIA_SOS.1 Verification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet **the minimum length established by the TOE for each authentication secret**.

5.1.4.5. FIA_UAU.1 Timing of authentication

FIA_UAU.1.1 The TSF shall allow **the following actions** on behalf of the user to be performed before the user is authenticated:

- **Perform start-up, self-test (FPT_TST.1), detection of the secure blocking state (FPT_FLS.1), detection of violation of physical integrity (FPT_PHP.2),**
- **Perform basic diagnostic functions, such as checking the communications from the host to the card, checking firmware level and token info and checking information on mechanisms supported.**
- **Open a session**
- **Access Public data objects**
- **Identification (FIA_UID.1).**

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

5.1.4.6. FIA_UID.1 Timing of identification

FIA_UID.1.1 The TSF shall allow **the following actions** on behalf of the user to be performed before the user is identified:

- **Perform start-up, self-test (FPT_TST.1), detection of the secure blocking state (FPT_FLS.1), detection of violation of physical integrity (FPT_PHP.2),**
- **Perform basic diagnostic functions, such as checking the communications from the host to the card, checking firmware level and token info and checking information on mechanisms supported.**

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.

5.1.5. Security management (FMT)

5.1.5.1. FMT_MSA.1 (ROLE_CRYPTO) Management of security attributes

FMT_MSA.1.1 (ROLE_CRYPTO) The TSF shall enforce the **Backup SFP and Crypto-SFP** to restrict the ability to **query, modify and delete** [assignment: **none**] the security attributes **Role Crypto User and Role Crypto Officer** to **Crypto Officer**.

5.1.5.2. FMT_MSA.1 (ROLE_AUDIT) Management of security attributes

FMT_MSA.1.1 (ROLE_AUDIT) The TSF shall enforce the **Audit-SFP** to restrict the ability to **query, modify and delete** [assignment: **none**] the security attributes **Role Auditor** to **Auditor**.

5.1.5.3. FMT_MSA.2 Secure security attributes

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for security attributes.

5.1.5.4. FMT_MSA.3 Static attribute initialization

FMT_MSA.3.1 The TSF shall enforce the **Audit SFP, Backup SFP and Crypto SFP** to provide restrictive default values for security attributes that are used to enforce the SFP.

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

5.1.5.5. FMT_MTD.1 (Access Control) Management of TSF data

FMT_MTD.1.1 (ACCESS_CONTROL) The TSF shall restrict the ability to query *and* modify the **access control lists** to **Crypto-officer**.

Application note:

The Crypto-officer is allowed to change the access control lists only within the limits of the defined roles.

Application Note: This SFR is included because it is required by the PP. It is, however, satisfied by default because the TOE does not implement access control lists. Access is granted only to the owner of an object as specified in FDP_ACF.1 (TAC).

5.1.5.6. FMT_MTD.1 (USER_Crypto) Management of TSF data

FMT_MTD.1.1 (USER_CRYPTO) The TSF shall restrict the ability to change default *and* delete the **Identity and RAD for user with role attribute Crypto Officer and Crypto User** to ***Security Officer***.

Application Note: The Security Officer is a more restrictive role than Crypto Officer as specified in the PP for the purposes of change default and delete Identity and RAD.

5.1.5.7. FMT_MTD.1 (USER_Audit) Management of TSF data

FMT_MTD.1.1 (USER_AUDIT) The TSF shall restrict the ability to change default *and* delete the **Identity and RAD for user with role attribute Auditor** to **Auditor**.[21]

5.1.5.8. FMT_MTD.1 (RAD) Management of TSF data

FMT_MTD.1.1 (RAD) The TSF shall restrict the ability to modify the **RAD** to ***Crypto Officer for the identity associated with the* RAD**.

5.1.5.9. FMT_MTD.1 (AUDIT) Management of TSF data

FMT_MTD.1.1 (AUDIT) The TSF shall restrict the ability to query the **audit data of the TSF required by FAU_GEN.1** to **Crypto Officer and Auditor**.

Application Note: The Crypto Officer role has been added to the roles capable of performing the specified operations because the SFR as stated in the PP contradicts O.Audit_CM and the SFR has been adapted to agree with the Objective.

5.1.5.10. FMT_SMF.1 Specification of Management Functions

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

1. **User management (FMT_MSA.1/ROLE_CRYPTO, FMT_MSA.1/ROLE_AUDIT, FMT_MTD.1/RAD, FMT_MTD.1/USER_CRYPTO and FMT_MTD.1/USER_AUDIT),**
2. **Management of audit data (FMT_MSA.3, FMT_MTD.1/AUDIT),**
3. **Management of TSF data (FMT_MTD.1/ACCESS_CONTROL).**

5.1.5.11. FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles **Security Officer, Crypto Officer, Crypto User and Auditor**.

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

[21] The Auditor role is held by the TSF.

Application note:

The Crypto Officer and Crypto User roles may be associated with only one user – the client application. The client application in the TOE environment may act as agent for more than one user demanding signing of DTBS by the HSM.

In the TOE, the Auditor role is held by the TSF.

5.1.6. Protection of the TOE Security Functions (FPT)

5.1.6.1. FPT_AMT.1 Abstract machine testing

FPT_AMT.1.1 The TSF shall run a suite of tests during initial start-up *and* at the request of an authorized user to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

Application note:

Even though the CMCSOB PP includes requirements to the hardware as physical protection, the TOE might not include all hardware of the cryptographic module. The TSF shall perform testing to demonstrate the security assumptions made about the underlying abstract machine upon which the TSF relies. This "abstract" machine could be a hardware/firmware platform, or it could be some known and assessed hardware/software combination acting as a virtual machine. An example of a security assumption is memory management unit providing support for information flow control (as required by FDP_IFF.4) or access control (as required by FDP_ACF.1 (AUDIT)) in the TOE.

5.1.6.2. FPT_FLS.1 Failure with preservation of secure state

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: **failures detected by the TSF FPT_AMT.1 and FPT_TST.1**.

Refined by adding:

The TSF shall destroy plaintext CSP-SCD and other secret and private keys if failures occur.

5.1.6.3. FPT_ITC.1 Inter-TSF confidentiality during transmission

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

Application note:

The SFR FPT_ITC.1 addresses the confidentiality protection of the TSF data if they are exported as part of the backup data.

5.1.6.4. FPT_ITI.1 Inter-TSF detection of modification

FPT_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote trusted IT product within the following metric: ~~**cryptographic checksum according to the list of approved algorithms and parameters**~~ **SHA-1 digest**.

FPT_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and perform ~~**alarm**~~ **error indication**[22] **to the Crypto Officer** if modifications are detected.

[22] The TSF notifies the Crypto Officer by means of error indicators rather than aural or visual alarm signals.

Application note:

The SFR FPT_ITI.1 addresses the integrity protection of the TSF data if they are imported as part of the backup data.

The SHA-1 checksum is calculated by the TSF and by the equivalent function within the backup token. It can be compared before and after backup and before and after recovery to validate integrity. Because it is done within the TSF and backup token, there is no chance of substituting an illegitimate key value and corresponding digest value. A keyed hash or digital signature is, therefore, not required.

FPT_ITC.1 and FPT_ITI.1 are included because they are specified by the PP. However, the cloning mechanism used to implement backup and restoration securely transfers individual key objects and TSF data is limited to a few object attributes used in the enforcement of the access control policy. The requirement for confidentiality and integrity protection of the backup data is best stated by FTP_ITC.1, which is included as an addition to the PP.

5.1.6.5. FPT_PHP.2 Notification of physical attack

FPT_PHP.2.1 The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

FPT_PHP.2.2 The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

FPT_PHP.2.3 For **TOE**, the TSF shall monitor the devices and elements and notify **local user** when physical tampering with the TSF's devices or TSF's elements has occurred.

Refined by adding:

The TSF shall detect physical tampering performed by opening the device or removal of a cover. Note that the TOE does not have a door or removable cover. Opening the TOE would, therefore, require the use of tools that would necessitate the removal of the TOE from its intended operating environment.

Application Note:

The notification about detected physical attacks may be given e.g. through functional interfaces (stopping any other services but alarm signalisation), acoustic or optic signals. The TOE non-IT environment should ensure that notification about physical tampering attempts given by the TOE shall be noticed by the CSP security personnel.

5.1.6.6. FPT_PHP.3 Resistance to physical attack

FPT_PHP.3.1 The TSF shall resist **physical tampering by opening the device or removal of a cover** to the **components which:**

- **generate CSP-SCD (FCS_CKM.1),**
- **create the signature with CSP-SCD (FCS_COP.1),**
- **store CSP-SCD,**
- **store other secret or private keys**

by responding automatically such that the TSP is not violated.

Refined by adding:

The TSF shall resist the tampering by destruction of plaintext CSP-SCD and other confidential secret and private keys if physical tampering performed by opening the device or removal of a cover is detected.

Application Note:

The TOE shall protect the confidentiality of the CSP-CSD and other secret and private keys in case of physical maintenance or physical tampering. If the detection of opening the device or removal of a cover might not be effective for the switched off device the TOE will destroy the CSP-SCD in case of loss of power. The TOE will invoke the TSF required by FCS_CKM.4 to destroy the CSP-SCD and all other plaintext secret and private keys. The destruction of the CSP-SCD will prevent the use of an attacked TOE for signing until restoring the operational state

5.1.6.7. FPT_RCV.1 Manual recovery

FPT_RCV.1.1 After **a failure or service discontinuity**[23], the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

5.1.6.8. FPT_TST.1 TSF testing

FPT_TST.1.1 The TSF shall run a suite of self-tests during initial start-up *and* at the request of the authorised user to demonstrate the correct operation of the TSF.[24]

FPT_TST.1.2 The TSF shall provide authorized users with the capability to verify the integrity of TSF data.

FPT_TST.1.3 The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code.

Refined by adding:

The TSF shall perform the self-tests as follows:

Initialisation

Extended firmware integrity test.

Power-Up and On-request Tests

Firmware integrity test,

Internal TSF data integrity test,

Cryptographic algorithm test,

Random number generator tests

Critical functions test.,

Conditional Tests

Pair-wise consistency test (for public and private keys),

Continuous random number generator test

[23] For the TOE, failure in this context refers to self-test failure. Any other failure would be catastrophic, leaving the TOE in a secure but non-recoverable state. In the case of a service discontinuity, the module will always return to service in a secure state. The details of its operational state when it returns to service are determined by the configurable policy set by the SO.

[24] The selection "at the conditions installation and maintenance" has been removed because there are no states of the TOE that correspond to installation and maintenance.

Application note:

The TSF performs self-tests according to FPT_TST.1 to ensure that the TOE is functioning properly. The extended software/firmware integrity test might verify error detecting codes, cryptographic checksums or digital signatures generated by the software/firmware developer or by other authorities. A digital signature might prove that the firmware or software is part of the evaluated product. The power-up software/firmware integrity test and internal TSF data integrity test may detect modification of these data if the device was switched off. The tests may be implemented by internally generated error detecting codes, cryptographic checksums or digital signatures. The cryptographic algorithm test may detect errors in hardware, firmware or software implementing critical cryptographic mechanisms (see FCS_CKM.1, FCS_COP.1/SIGN). The test might be a known-answer-test (e.g. for encryption) or a pair-wise consistency test (e.g. verifying a generated signature before the signature is exported). Supplementary tests shall detect error of the random number generator used for the generation of CSP-SCD (see FCS_CKM.1 and FCS_RND.1), cryptographic keys or parameters. If any critical function is not covered by these tests the TSF should implement additional self-tests. The pair-wise consistency test for public and private keys may detect errors in the key generation process. Other consistency tests may check the correctness of the signing process and other cryptographic processes to prevent e.g. differential fault attacks. Manual key entry test may detect errors to prevent use of incorrect keys if manual key entry is implemented. Continuous random number generator test may detect failure in operation of the generator to prevent use of wrong random number.

The TOE shall verify the integrity and authenticity of the TSF executable code at installation, maintenance and initialisation to prevent malicious software running on the TOE.

5.1.7. Trusted path (FTP)

5.1.7.1. FTP_TRP.1 Trusted path

FTP_TRP.1.1 The TSF shall provide a communication path between itself and local users that is *physically and* logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

FTP_TRP.1.2 The TSF shall permit local users to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for initial user authentication *(FIA_UID.1, FIA_UAU.1)*, **TSF management (FMT_MSA.1/ROLE, FMT_MTD.1/USER_CRYPTO, FMT_MTD.1/USER_AUDIT, FMT_MTD.1/RAD, FMT_MSA.2, FMT_MSA.3, FMT_MTD.1/ACCESS, FMT_MTD.1/AUDIT, FMT_SMR.1)** *and* the following additional functions:

- **Upload of the TSF-generated authentication data to the iKey**
- **Upload of the TSF-generated challenge secret to the PED display**
- **Entry of M of N Activation secret shares**
- **Entry of the Token Cloning Domain key**

5.2. Additions to the PP

For ease of identification, all TOE security functional requirements additional to the PP are collected in this section.

5.2.1. Cryptographic support (FCS)

5.2.1.1. FCS_CKM.2 (FW Update) Cryptographic key distribution

FCS_CKM.2.1 (FW Update) The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **key exchange** that meets the following: **Luna® firmware update protocol**.

5.2.1.2. FCS_CKM.3 Cryptographic key access

FCS_CKM.3.1(Access) The TSF shall perform **key access** in accordance with a specified cryptographic key access method, **return of a key handle,** that meets the following: **PKCS #11 standard**.

5.2.1.3. FCS_COP.1 (DIGEST) Cryptographic operation - Message digest

FCS_COP.1.1 (DIGEST) The TSF shall perform **message digest** in accordance with *the* specified cryptographic *algorithms* **listed below**:

(1) SHA-1 (FIPS PUB 180-2),

(2) SHA-256 (FIPS PUB 180-2),

(3) SHA-384 (FIPS PUB 180-2),

(4) SHA-512 (FIPS PUB 180-2).

5.2.1.4. FCS_COP.1 (RSA ENC/DEC) Cryptographic operation - RSA Encrypt/Decrypt

FCS_COP.1.1 (RSA ENC/DEC) The TSF shall perform **asymmetric encryption and decryption** in accordance with a specified cryptographic algorithm **RSA** and cryptographic key sizes **1024 bits, 2048 bits and 4096 bits** that meet the following: **PKCS #1 V1.5 and OAEP**.

5.2.1.5. FCS_COP.1 (TDES ENC/DEC) Cryptographic operation - TDES Encrypt/Decrypt

FCS_COP.1.1 (TDES Enc/Dec) The TSF shall perform **symmetric encryption and decryption** in accordance with a specified cryptographic algorithm **Triple DES (ECB and CBC mode)** and cryptographic key sizes **112 bits, and 168 bits** that meet the following: **FIPS PUB 46-3**.

5.2.1.6. FCS_COP.1 (AES ENC/DEC) Cryptographic operation - AES Encrypt, Decrypt

FCS_COP.1.1 (AES Enc/Dec) The TSF shall perform **symmetric encryption and decryption** in accordance with a specified cryptographic algorithm **AES (ECB and CBC mode)** and cryptographic key sizes **128 bits, 192 bits and 256 bits** that meet the following: **FIPS PUB 197**.

5.2.2. User data protection (FDP)

5.2.2.1. FDP_ACC.1 (TAC) Subset access control

FDP_ACC.1.1 (TAC) The TSF shall enforce the **Token Access Control (TAC) SFP** on **the following:**

(1) Device sessions (subjects).

(2) Private keys, public keys, secret keys, certificates, data objects.

(3) Operations:
- **a. Read (Query Attribute Value)**
- **b. Modify**
- **c. Destroy**
- **d. Generate**[25]
- **e. Wrap (export)**
- **f. Use**[26]
- **g. Clone**

[25] The Generate operation is intended primarily to indicate symmetric key or asymmetric key pair generation. However, it also includes other methods of creating an object in the TOE, such as importing (unwrapping) a key and generic data object creation.
[26] The Use operation includes symmetric key encryption/decryption, private key signing and decryption, and public key verification and encryption.

5.2.2.2. FDP_ACF.1 (TAC) Security attribute based access control

FDP_ACF.1.1 (TAC) The TSF shall enforce the **Token Access Control (TAC) SFP** to objects based on **the following:**

(1) Subject attributes:
- **a. Session and Access ID**
- **b. User ID associated with session (Access Owner)**
- **c. Role.**

(2) Object attributes:
- **a. Private. If True, object is Private. If False, object is Public.**
- **b. Owner. Object ownership is assigned to the object creator.**
- **c. Sensitive. If True, object is Sensitive. If False, object is Non-Sensitive.**
- **d. Extractable. If True, object may be extracted. If False, object may not be extracted.**
- **e. Modifiable. If True, object may be modified. If False, object may not be modified.**
- **f. Type. Public key, private key, secret key.**

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

A subject may perform an allowed operation on an object if one of the following two conditions holds:

(1) The object is a "Public" object, i.e., the PRIVATE attribute is FALSE, or

(2) The User ID of the subject is the same as the object's owner.

An allowed operation is one permitted by the object attribute definitions within the constraints of the HSM and Partition level capability and policy settings. Table 5-2 summarizes the operations allowed by the object attribute settings.

The operations allowed for each user type in Table 5-2 have been abbreviated as indicated in Table 5-1 below.

Table 5-1 – Operation Abbreviations

Operation	Abbreviation	Operation	Abbreviation
Clone	C	Read	R
Destroy	D	Use	U
Generate	G	Wrap	W
Modify	M		

Table 5-2 – Access Matrix

Object Attribute				Subject (ID/Role)		
Private	**Sensitive**	**Modifiable**	**Extractable**	**User ID/CO**	**User ID/CU**	**Public/Nil**
0	0	0	N/A	C,D,G,R	R	D,G,R
0	0	1	N/A	C,D,G,M,R	R	D,G,M,R
1	0	0	N/A	C,D,G,R	R	---
1	0	1	N/A	C,D,G,M,R	R	---
1	1	0	0	C,D,G,R(1),U	R(1),U	---
1	1	1	0	C,D,G,M,R(1),U	R(1),U	---
1	1	0	1	C,D,G,R(1),U,W	R(1),U	---
1	1	1	1	C,D,G,M,R(1),U,W	R(1),U	---

1. The plaintext value of key material stored in objects whose CKA_SENSITIVE attribute is set cannot be read although other object values, such as the object label, are accessible.
2. A "0" in the **Table 5-2** indicates that the attribute labelling the column has not been set. A "1" indicates that the attribute has been set.

FDP_ACF.1.3 (TAC) The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: **None**.

FDP_ACF.1.4 (TAC) The TSF shall explicitly deny access of subjects to objects based on the **following additional rules**:

- **A subject shall not have access to the plaintext value of an object whose CKA_SENSITIVE attribute is set.**

5.2.2.3. FDP_DAU.1 Basic data authentication

FDP_DAU.1.1 The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of **user objects**.

FDP_DAU.1.2 The TSF shall provide the **Users** with the ability to verify evidence of the validity of the indicated information.

Refined by adding:

The evidence is provided by the SHA-1 fingerprint of the object. It is generated by the TOE and can be queried at any time by the user.

5.2.2.4. FDP_DAU.2 Data authentication with identity of guarantor

FDP_DAU.2.1 The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of **private and public keys**.

FDP_DAU.2.2 The TSF shall provide the **Users** with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

Refined by adding:

The evidence is provided by the Public Key Confirmation (PKC). It is generated in an X.509 certificate format by the TOE and is signed by either the internal Hardware Origin Key, whose public key is certified by the SafeNet trust anchor or by a customer private key, whose public key certificate has been issued by a third party CSP or Trust Centre.

5.2.2.5. FDP_ITC.1 Import of user data without security attributes

FDP_ITC.1.1 The TSF shall enforce the **Token Access Control (TAC) SFP** when importing user data controlled under the SFP from outside of the TSC.

FDP_ITC.1.2 The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC.

FDP_ITC.1.3 The TSF shall enforce the following rules when importing user data, controlled under the SFP, from outside the TSC*:* **The CKA_SENSITIVE attribute of the object storing user data imported via an Unwrap operation shall be set.**

5.2.2.6. FDP_RIP.2 Full residual information protection

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

5.2.2.7. FDP_UCT.1 Basic data exchange confidentiality

FDP_UCT.1.1 The TSF shall enforce the **Token Access Control (TAC) SFP** to be able to transmit *and* receive objects in a manner protected from unauthorised disclosure.

5.2.2.8. FDP_UIT.1 Data exchange integrity

FDP_UIT.1.1 The TSF shall enforce the **Token Access Control (TAC) SFP** to be able to transmit *and* receive user data in a manner protected from modification errors.

FDP_UIT.1.2 The TSF shall be able to determine on receipt of user data, whether modification has occurred.

5.2.3. Identification and authentication (FIA)

5.2.3.1. FIA_SOS.2 TSF generation of secrets

FIA_SOS.2.1 The TSF shall provide a mechanism to generate secrets that meet **the minimum lengths for each function for which they are required and that are random:**

- **SO and User PED authentication data 48 bytes**
- **User Challenge secret 75 bits**
- **M of N activation 32 bytes**
- **Cloning 24 bytes.**

FIA_SOS.2.2 The TSF shall be able to enforce the use of TSF generated secrets for **the following TSF functions:**

- **SO and User PED authentication**
- **M of N activation**
- **Cloning.**

5.2.3.2. FIA_UAU.4 Single-use authentication mechanisms

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to **Challenge-response authentication for Users**.

5.2.3.3. FIA_UAU.5 Multiple authentication mechanisms

FIA_UAU.5.1 The TSF shall provide **the following authentication mechanisms** to support user authentication:

- **M of N secret sharing (SO and User)**.

- **PED Key entry (SO and User)**
- **PED PIN entry (SO and User)**
- **Challenge-response (User)**

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to **the following rules:**

- **The user must enter their authentication data using, at a minimum, the PED and an iKey.**
- **Optionally, the user enters a Personal Identification Number (PIN) via the PED in addition to the entering the iKey.**
- **If required by the policy defined for the TOE, M out of N secret shares must first be entered via the Luna® PIN Entry Device (PED) in order to enable the TOE for operation.**
- **The User, in the Crypto User and Crypto Officer roles, must enter the challenge secret corresponding to one of the roles via the application interface in order to access cryptographic data and services.**

5.2.3.4. FIA_USB.1 User-subject binding

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

- **User ID**
- **User checkword**
- **User role.**

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

- **User ID is Public (unidentified)**
- **User checkword is Nil.**
- **User role is Nil.**

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

When the user is successfully authenticated the user security attributes change from their initial values to the values appropriate for that authenticated user.

5.2.4. Security management (FMT)

5.2.4.1. FMT_MOF.1 Management of security functions behaviour

FMT_MOF.1.1 The TSF shall restrict the ability to disable, enable *and* modify the behaviour of the functions **listed below** to **the Security Officer role:**

HSM Level

M of N Activation – SO may enable and disable.
M of N Auto-activation – SO may enable and disable.
HSM Cloning – SO may enable and disable.
Remote Authentication – SO may enable and disable.
Network Replication – SO may enable and disable.
Force change of User authentication data – SO may enable and disable.

Partition Level

Partition reset – SO may enable and disable.

Partition activation – SO may enable and disable.

Partition auto-activation – SO may enable and disable.

High Availability – SO may enable and disable.

Multi-purpose keys – SO may enable and disable.

Changing key attributes once a key has been created – SO may enable and disable.

Operation without RSA blinding – SO may enable and disable.

Signing operations with non-local keys – SO may enable and disable.

Performing raw RSA operations – SO may enable and disable.

Private key unwrapping – SO may enable and disable.

Secret key wrapping – SO may enable and disable.

Secret key unwrapping – SO may enable and disable.

User key management capability – SO may enable and disable.

Increment failed login attempt counter on failed challenge response validation – SO may enable and disable.

RSA signing without confirmation – SO may enable and disable.

5.2.4.2. FMT_MSA.1 (Object Attributes) Management of security attributes

FMT_MSA.1.1 (Object Attributes) The TSF shall enforce the **TAC SFP** to restrict the ability to modify the security attributes **CKA_PRIVATE (for data and certificate objects only), CKA_EXTRACTABLE (for secret keys only), CKA_DERIVE (for secret keys only) and CKA_MODIFIABLE** to **the Crypto Officer role.**

5.2.4.3. FMT_MSA.2 (Object Attributes) Secure security attributes

FMT_MSA.2.1 (Object Attributes) The TSF shall ensure that only secure values are accepted for security attributes.

5.2.4.4. FMT_MSA.3 (Object Attributes) Static attribute initialization

FMT_MSA.3.1 (Object Attributes) The TSF shall enforce the **TAC SFP** to provide restrictive default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 (Object Attributes) The TSF shall allow **Crypto Officer** to specify alternative initial values to override the default values when an object or information is created.

5.2.4.5. FMT_MTD.1 (Login Failures) Management of TSF data

FMT_MTD.1.1 (Login Failures) The TSF shall restrict the ability to change_default the **Number of User Login Failures Allowed (FIA_AFL.1.1 (User))** to the **Security Officer.**

5.2.4.6. FMT_MTD.1 (UAV) Management of TSF data

FMT_MTD.1.1 (UAV – User Locked Flag) The TSF shall restrict the ability to change_default, query, modify *and* delete the **User Locked Flag** to **the Security Officer role.**

FMT_MTD.1.1 (Challenge Secret – Crypto Officer) The TSF shall restrict the ability to modify the **Crypto Officer Challenge Secret** to **the Crypto Officer role.**

5.2.4.7. FMT_MTD.1 (SOV) Management of TSF data

FMT_MTD.1.1 (SOV) The TSF shall restrict the ability to change_default *and* modify the **SO Checkword** to **the Security Officer role.**

5.2.4.8. FMT_SMF.1 (Policies) Specification of Management Functions

FMT_SMF.1.1 (Policies) The TSF shall be capable of performing the following security management functions:

1. **disable, enable and modify the behaviour of configurable policy settings at the HSM and Partition levels (FMT_MOF.1),**
2. **change_default the Number of User Login Failures Allowed.**

5.2.5. Protection of the TOE Security Functions (FPT)

5.2.5.1. FPT_RVM.1 Non-bypassability of the TSP

FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

5.2.5.2. FPT_SEP.1 TSF domain separation

FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

5.2.6. Resource utilization (FRU)

5.2.6.1. FRU_FLT.1 Degraded fault tolerance

FRU_FLT.1.1 The TSF shall ensure the operation of **TOE's user data protection capabilities** when the following failures occur: **power failure or data input/output failure**.

5.2.7. Trusted path (FTP)

5.2.7.1. FTP_ITC.1 (FW Update) Inter-TSF trusted channel

FTP_ITC.1.1 (FW Update) The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 (FW Update) The TSF shall permit the remote trusted IT product to initiate communication via the trusted channel.

FTP_ITC.1.3 (FW Update) The TSF shall initiate communication via the trusted channel for **firmware load and update**.

5.2.7.2. FTP_ITC.1 (Key Cloning) Inter-TSF trusted channel

FTP_ITC.1.1 (Key Cloning) The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 (Key Cloning) The TSF shall permit the TSF, the remote trusted IT product to initiate communication via the trusted channel.

FTP_ITC.1.3 (Key Cloning) The TSF shall initiate communication via the trusted channel for **key cloning**.

Application Note:

The function specified by FTP_ITC.1 (Key Cloning) is used to establish a trusted channel between the TOE and the Backup Token. The session key used to encrypt the data being backed up or restored is negotiated as part of the establishment of the trusted channel.

5.3. Requirements for IT Environment

5.3.1. Security audit (FAU)

5.3.1.1. FAU_SAR.1 (ENV) Audit review

FAU_SAR.1.1 (ENV) The *IT environment* shall provide **System auditor of the CSP** with the capability to read **all audit information produced by the TOE** from the audit records.

FAU_SAR.1.2 (ENV) The *IT environment* shall provide the audit records in a manner suitable for the user to interpret the information.

5.3.1.2. FAU_STG.1 (ENV) Guarantees of audit data availability

FAU_STG.1.1 (ENV) The *IT environment* shall protect the stored audit records from unauthorised deletion.

FAU_STG.1.2 (ENV) The *IT environment* shall be able to **prevent** unauthorised modifications to the audit records in the audit trail.

Application note:

The SFR FAU_STG.1 (ENV) addresses the protection of the audit trail generated and exported by the TOE that is provided by the IT environment.

5.3.2. Cryptographic support (FCS)

5.3.2.1. FCS_CKM.1 (ENV/FW Update) Cryptographic key generation

FCS_CKM.1.1 (ENV/FW Update) The *firmware update application system* shall generate cryptographic keys in accordance with *the* specified cryptographic key generation *algorithms* **listed below** and specified cryptographic key sizes **specified for each algorithm** that meet the following **standards noted for each algorithm**:

RSA 2048 bits key pairs in accordance with ANSI X9.31.
TDES 112 bits in accordance with FIPS PUB 46-3 and ANSI X9.52.

5.3.2.2. FCS_CKM.2 (ENV/FW Update) Cryptographic key distribution

FCS_CKM.2.1 (ENV/FW Update) The *firmware update application system* shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **key exchange** that meets the following: **Luna® firmware update protocol**.

5.3.2.3. FCS_CKM.2 (ENV/BACKUP) Cryptographic key distribution

FCS_CKM.2.1 (ENV/BACKUP) The *Backup Token* shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **key negotiation** that meets the following: **RSA 4096 bits and TDES 168 bits**.

5.3.2.4. FCS_COP.1 (ENV/ENC FW Update) Cryptographic operation

FCS_COP.1.1 (ENV/ENC FW Update) The *firmware update application system* shall perform **encryption *and* decryption** in accordance with a specified cryptographic algorithm **TDES** and cryptographic key sizes **112 bits** that meet the following: **ANSI X9.52**.

5.3.2.5. FCS_COP.1 (ENV/SIGN FW Update) Cryptographic operation

FCS_COP.1.1 (ENV/SIGN FW Update) The *firmware update application system* shall perform **digital signature generation** in accordance with a specified cryptographic algorithm **RSA with SHA-1** and cryptographic key sizes **1024 bits** that meet the following: **standards noted for each algorithm**.

RSA – PKCS #1 V1.5

SHA-1 – FIPS PUB 180-2.

5.3.2.6. FCS_COP.1 (ENV/BACKUP_ENC) Cryptographic operation

FCS_COP.1.1 (ENV/BACKUP_ENC) The *Backup Token* shall perform **encryption *and* decryption** in accordance with a specified cryptographic algorithm **TDES** and cryptographic key sizes **168 bits** that meet the following: **FIPS PUB 46-3**.

5.3.2.7. FCS_COP.1 (ENV/BACKUP_INT) Cryptographic operation

FCS_COP.1.1 (ENV/BACKUP_INT) The *Backup Token* shall perform **calculation and verification of cryptographic checksums** in accordance with a specified cryptographic algorithm **SHA-1** and cryptographic key sizes **N/A** that meet the following: **FIPS PUB 180-2**.

5.3.3. User data protection (FDP)

The client application shall provide the TOE signing function to its authorised end-user only and shall prevent unauthorised transmission and manipulation of DTBS representation to be signed by the TOE.

5.3.3.1. FDP_ACC.1 (CLIENT) Subset access control

FDP_ACC.1.1 (CLIENT) The *IT environment* shall enforce the **Client application SFP** on **end-user, Cryptographic module signing function, use**.

5.3.3.2. FDP_ACF.1 (CLIENT) Security attribute based access control

FDP_ACF.1.1 (CLIENT) The *IT environment* shall enforce the **Client application SFP** to **objects** based on **authorisation for Cryptographic module signing function**.

FDP_ACF.1.2 (CLIENT) The *IT environment* shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **authorised end-user is allowed to use Cryptographic module signing function**.

FDP_ACF.1.3 (CLIENT) The *IT environment* shall explicitly authorise access of subjects to objects based on the following additional rules: **none**.

FDP_ACF.1.4 (CLIENT) The IT environment shall explicitly deny access of subjects to objects based on the rule: **non-authorised end-user is not allowed to use Cryptographic module signing function**.

Application Note:

The security attribute “authorisation for Cryptographic module signing function” is assigned to end-users of the client application with two possible values:

(a) authorised to use Cryptographic module signing function,

(b) not authorised to use Cryptographic module signing function.

5.3.3.3. FDP_ACC.1 (ENV/BACKUP) Subset access control

FDP_ACC.1.1 (ENV/BACKUP) The *Backup Token* shall enforce the **Backup Token SFP** on **User; CSP-SCD, backup key(s), backup data; backup (FDP_BKP.1), restore (FDP_BKP.1), backup key distribution (FCS_CKM.2)**.

5.3.3.4. FDP_ACF.1 (ENV/BACKUP) Security attribute based access control

FDP_ACF.1.1 (ENV/BACKUP) The *Backup Token* shall enforce the **Backup Token SFP** to objects based on **Identity and Role**.

FDP_ACF.1.2 (ENV/BACKUP) The *Backup Token* shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **User with security attribute Role Crypto Officer is allowed to restore CSP-SCD and CSP-SVD (FDP_BKP.1) to the TOE.**

FDP_ACF.1.3 (ENV/BACKUP) The *Backup Token* shall explicitly authorise access of subjects to objects based on the following additional rules: **User with security attribute Role Crypto Officer is allowed to restore all partition objects to a partition in the TOE.**

FDP_ACF.1.4 (ENV/BACKUP) The *Backup Token* shall explicitly deny access of subjects to objects based on the **none**

5.3.3.5. FDP_UIT.1 Data exchange integrity

FDP_UIT.1.1 The *IT environment* shall enforce the **Client application SFP** to be able to transmit user data in a manner protected from modification, deletion *and* insertion errors.

FDP_UIT.1.2 The *IT environment* shall be able to determine on receipt of user data, whether modification, deletion *and* insertion has occurred.

Application note:

The user data to be protected by the IT environment are data to be signed by the Cryptographic module.

5.3.4. Identification and authentication (FIA)

The client application shall identify and authenticate its end-user for use of the Cryptographic module services.

5.3.4.1. FIA_UAU.1 (CLIENT) Timing of authentication

FIA_UAU.1.1 (CLIENT) The *IT environment* shall allow **user login** on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 (CLIENT) The *IT environment* shall require each user to be successfully authenticated before allowing any other actions on behalf of that user.

5.3.4.2. FIA_UID.1 (CLIENT) Timing of identification

FIA_UID.1.1 (CLIENT) The *IT environment* shall allow **user login** on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 (CLIENT) The *IT environment* shall require each user to be successfully identified before allowing any other actions of the IT environment on behalf of that user.

5.3.5. Trusted path (FTP)

5.3.5.1. FTP_TRP.1 (CLIENT) Trusted path

FTP_TRP.1.1 (CLIENT) The *IT environment* shall provide a communication path between itself and local users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

FTP_TRP.1.2 (CLIENT) The IT environment shall permit local users to initiate communication via the trusted path.

FTP_TRP.1.3 (CLIENT) The IT environment shall require the use of the trusted path for communication with TOE for identification, authentication *and* management.

Application note:

The Cryptographic module does provide a human user interface for authentication and management on the host system. The client application will provide this interface and a trusted path for the communication between the user and the Cryptographic module. The client application shall support the trusted path as one for the communication entity.

5.3.6. Trusted path (FTP)

5.3.6.1. FTP_ITC.1 (ENV/FW Update) Inter-TSF trusted channel

FTP_ITC.1.1 (ENV/FW Update) The *firmware update application system* shall provide a communication channel between itself and *a* remote *Luna PCI* that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 (ENV/FW Update) The *firmware update application system* shall permit the *firmware update application Luna PCI* to initiate communication via the trusted channel.

FTP_ITC.1.3 (ENV/FW Update) The *firmware update application system* shall initiate communication via the trusted channel for *firmware load and update*.

5.3.6.2. FTP_ITC.1 (ENV/Key Cloning) Inter-TSF trusted channel

FTP_ITC.1.1 (ENV/Key Cloning) The *Backup Token* shall provide a communication channel between itself and a remote trusted *Luna PCI* that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 (ENV/Key Cloning) The *Backup Token* shall permit *the Backup Token*, the remote trusted *Luna PCI* to initiate communication via the trusted channel.

FTP_ITC.1.3 (ENV/Key Cloning) The *Backup Token* shall initiate communication via the trusted channel for **key cloning**.

5.4. Non-IT Requirements

RE.ENV_Outage_Protection

The CSP personnel shall ensure that the power supplied to the TOE is adequately protected against unexpected interruptions and the effects of surges and voltage fluctuations outside the normal operating range of the device and that the TOE is operated in an environment that is provided adequate protection against disasters such as fire and flood.

RE.ENV_Personnel Personnel security measures

The CSP shall define the obligations and the services of management and operation roles for the TOE. The CSP shall inform and train the personnel for their roles. The CSP shall inform the personnel using the TOE about their civil, financial and legal responsibilities.

RE.ENV_Protect_Access Physical protection of the TOE

The CSP shall establish physical and organisational security measures to protect the TOE against modification of TOE hardware, firmware and software. These measures shall restrict the access to the TOE and protected assets to authorised persons. If the TOE detects and notifies about physical tampering the local users shall inform the CSP security staff. The TOE shall not be used until the physical integrity of the TOE is established.

RE.ENV_Recovery Recovery procedures for the TOE

The CSP shall define and apply recovery plans and procedures which allow a secure and timely recovery of the TOE operational state. These procedures shall ensure at least

(1) secure initialisation of new TOE devices replacing other TOE devices,

(2) re-initialisation of TOE devices establishing the secure state by the TSF FPT_FLS.1 after detecting failures by the TSF FPT_AMT.1 and FPT_TST.1,

(3) integrity check of the TOE hardware, firmware and software and re-initialisation of TOE devices if the TOE indicates physical tampering by TSF FPT_PHP.2 and destroyed the plaintext SCP-SCD and other confidential secret and private keys by TSF FPT_PHP.3.

To support the TOE backup of the CSP-SCD, other user data and TSF data the CSP will ensure the availability of the backup data and the cryptographic quality, confidentiality and availability of the backup keys.

RE.ENV_Secure_Init Secure initialisation of the TOE

The CSP shall define and apply procedures and controls in the TOE environment which allow to securely set-up and initialise the TOE for the generation of CSP-SCD and signatures. This includes

(1) dual control for secure installation and initialisation of the TOE in the CSP,

(2) the CSP-SCD / CSP-SVD pair generation,

(3) the export of the CSP-SVD by the TOE and the securing the authenticity of the CSP-SVD,

(4) the secure initial configuration of the TSF data user's identity, roles and user authentication information.

RE.ENV_Secure_Oper Secure operation of the TOE

The CSP shall define and apply procedures and controls in the TOE environment which allow operating the TOE within a CA system in compliance with the requirements of the EU directive, the Qualified Certificates Policy for the issued certificates, the secure operation of the client application and the TOE guidance. The TOE user shall ensure that notification about physical tampering attempts given by the TOE will be noticed by the CSP security personnel.

5.5. TOE Security Assurance Requirements

The assurance requirements for this TOE are as specified in the Common Criteria Version 2.3 Part 3-EAL 4 package with augmentation. The EAL 4 package has been augmented by the addition of the Part 3 requirements: ADV_IMP.2, ALC_FLR.2, AVA_CCA.1, AVA_MSU.3, and AVA_VLA.4.

5.5.1. Security Assurance Requirements Augmentation to EAL 4

5.5.1.1. ADV_IMP.2 Implementation of the TSF

Dependencies: ADV_LLD.1 Descriptive low-level design

ALC_TAT.1 Well-defined development tools

Application notes

The ADV_IMP.2.2E element defines a requirement that the evaluator determine that the implementation representation is an accurate and complete instantiation of the TOE security functional requirements. This provides a direct correspondence between the TOE security functional requirements and the implementation representation, in addition to the pairwise correspondences required by the Representation correspondence (ADV_RCR) family. It is expected that the evaluator will use the evidence provided in Representation correspondence (ADV_RCR) as an input to making this determination.

Developer action elements:

ADV_IMP.2.1D The developer shall provide the implementation representation for the **entire** TSF.

Content and presentation of evidence elements:

ADV_IMP.2.1C The implementation representation shall unambiguously define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV_IMP.2.2C The implementation representation shall be internally consistent.

ADV_IMP.2.3C The implementation representation shall describe the relationships between all portions of the implementation.

Evaluator action elements:

ADV_IMP.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_IMP.2.2E The evaluator shall determine that the implementation representation is an accurate and complete instantiation of the TOE security functional requirements.

5.5.1.2. ALC_FLR.2 Flaw reporting procedures

Dependencies: No dependencies.

Objectives

In order for the developer to be able to act appropriately upon security flaw reports from TOE users, and to know to whom to send corrective fixes, TOE users need to understand how to submit security flaw reports to the developer. Flaw remediation guidance from the developer to the TOE user ensures that TOE users are aware of this important information.

Developer action elements:

ALC_FLR.2.1D The developer shall provide flaw remediation procedures addressed to TOE developers.

ALC_FLR.2.2D The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.

ALC_FLR.2.3D The developer shall provide flaw remediation guidance addressed to TOE users.

Content and presentation of evidence elements:

ALC_FLR.2.1C The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

ALC_FLR.2.2C The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

ALC_FLR.2.3C The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

ALC_FLR.2.4C The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

ALC_FLR.2.5C The flaw remediation procedures shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.

ALC_FLR.2.6C The procedures for processing reported security flaws shall ensure that any reported flaws are corrected and the correction issued to TOE users.

ALC_FLR.2.7C The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.

ALC_FLR.2.8C The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.

Evaluator action elements:

ALC_FLR.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.5.1.3. Covert channel analysis (AVA_CCA.1)

Dependencies: ADV_FSP.2 Fully defined external interfaces

ADV_IMP.2 Implementation of the TSF

AGD_ADM.1 Administrator guidance

AGD_USR.1 User guidance

Objectives

The objective is to identify covert channels that are identifiable, through an informal search for covert channels.

Developer action elements:

AVA_CCA.1.1D The developer shall conduct a search for covert channels for each information flow control policy.

AVA_CCA.1.2D The developer shall provide covert channel analysis documentation.

Content and presentation of evidence elements:

AVA_CCA.1.1C The analysis documentation shall identify covert channels and estimate their capacity.

AVA_CCA.1.2C The analysis documentation shall describe the procedures used for determining the existence of covert channels, and the information needed to carry out the covert channel analysis.

AVA_CCA.1.3C The analysis documentation shall describe all assumptions made during the covert channel analysis.

AVA_CCA.1.4C The analysis documentation shall describe the method used for estimating channel capacity, based on worst case scenarios.

AVA_CCA.1.5C The analysis documentation shall describe the worst case exploitation scenario for each identified covert channel.

Evaluator action elements:

AVA_CCA.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_CCA.1.2E The evaluator shall confirm that the results of the covert channel analysis show that the TOE meets its functional requirements.

AVA_CCA.1.3E The evaluator shall selectively validate the covert channel analysis through testing.

5.5.1.4. Analysis and testing for insecure states (AVA_MSU.3)

Dependencies: ADO_IGS.1 Installation, generation, and start-up procedures

ADV_FSP.1 Informal functional specification

AGD_ADM.1 Administrator guidance

AGD_USR.1 User guidance

Objectives

The objective is to ensure that misleading, unreasonable and conflicting guidance is absent from the guidance documentation, and that secure procedures for all modes of operation have been addressed. Insecure states should be easy to detect. In this component, an analysis of the guidance documentation by the developer is required to provide additional assurance that the objective has been met, and this analysis is validated and confirmed through testing by the evaluator.

Application notes

In this component the evaluator is required to undertake testing to ensure that if and when the TOE enters an insecure state this may easily be detected. This testing may be considered as a specific aspect of penetration testing.

Developer action elements:

AVA_MSU.3.1D The developer shall provide guidance documentation.

AVA_MSU.3.2D The developer shall document an analysis of the guidance documentation.

Content and presentation of evidence elements:

AVA_MSU.3.1C The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AVA_MSU.3.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.3.3C The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.3.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

AVA_MSU.3.5C The analysis documentation shall demonstrate that the guidance documentation is complete.

Evaluator action elements:

AVA_MSU.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_MSU.3.2E The evaluator shall repeat all configuration and installation procedures, and other procedures selectively, to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.

AVA_MSU.3.3E The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

AVA_MSU.3.4E The evaluator shall confirm that the analysis documentation shows that guidance is provided for secure operation in all modes of operation of the TOE.

AVA_MSU.3.5E The evaluator shall perform independent testing to determine that an administrator or user, with an understanding of the guidance documentation, would reasonably be able to determine if the TOE is configured and operating in a manner that is insecure.

5.5.1.5. Highly resistant (AVA_VLA.4)

Dependencies: ADV_FSP.1 Informal functional specification

ADV_HLD.2 Security enforcing high-level design

ADV_IMP.1 Subset of the implementation of the TSF

ADV_LLD.1 Descriptive low-level design

AGD_ADM.1 Administrator guidance

AGD_USR.1 User guidance

Objectives

A vulnerability analysis is performed by the developer to ascertain the presence of security vulnerabilities, and to confirm that they cannot be exploited in the intended environment for the TOE.

The evaluator performs independent penetration testing, supported by the evaluator's independent vulnerability analysis, to determine that the TOE is resistant to penetration attacks performed by attackers possessing a high attack potential.

Developer action elements:

AVA_VLA.4.1D The developer shall perform a vulnerability analysis.

AVA_VLA.4.2D The developer shall provide vulnerability analysis documentation.

Content and presentation of evidence elements:

AVA_VLA.4.1C The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for ways in which a user can violate the TSP.

AVA_VLA.4.2C The vulnerability analysis documentation shall describe the disposition of identified vulnerabilities.

AVA_VLA.4.3C The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

AVA_VLA.4.4C The vulnerability analysis documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.

AVA_VLA.4.5C The vulnerability analysis documentation shall show that the search for vulnerabilities is systematic.

AVA_VLA.4.6C The vulnerability analysis documentation shall provide a justification that the analysis completely addresses the TOE deliverables.

Evaluator action elements:

AVA_VLA.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VLA.4.2E The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure the identified vulnerabilities have been addressed.

AVA_VLA.4.3E The evaluator shall perform an independent vulnerability analysis.

AVA_VLA.4.4E The evaluator shall perform independent penetration testing, based on the independent vulnerability analysis, to determine the exploitability of additional identified vulnerabilities in the intended environment.

AVA_VLA.4.5E The evaluator shall determine that the TOE is resistant to penetration attacks performed by an attacker possessing a high attack potential.

5.6. Strength of Function Claim

The minimum Strength of Function (SoF) required for the TOE is **SoF - high**. This applies to the following security functional requirements:

FCS_RND.1 Quality metrics for random numbers

FIA_UAU.1 Timing of authentication

FIA_UAU.4 Single-use authentication mechanisms

FIA_UAU.5 Multiple authentication mechanisms

FIA_SOS.2 TSF generation of secrets

FTP_ITC.1 Inter-TSF trusted channel (The cloning domain identifier is generated and used as part of the cloning protocol in authenticating the TOE and Backup Token as part of the same cloning domain and, therefore, allowed to participate in the trusted channel.)

This is considered to be necessary and reasonable for the TOE because its primary function of protecting user key material in all environments relies heavily on the authentication processes.

The assessment of algorithmic strength of cryptographic functions does not form part of the evaluation.

6. TOE SUMMARY SPECIFICATION

6.1. Overview

The TOE is used primarily as a hardware security module for the protection of the private signing keys at the Certification Authority (CA), or Certification Service Provider (CSP), within a Public Key Infrastructure (PKI). As such, its primary functions are to securely generate and protect the private signing key used by the CA or CSP when signing digital certificates.

The Luna® PCI cryptographic module provides storage capability for cryptographic material generated by the module or generated by the host application as well as storage for non-cryptographic data provided by the host application. Non-cryptographic data can be stored in the form of certificate objects or in the form of data objects. When storing or generating keys (secret or private), the module imposes some restrictions on how these keys are handled. Security policy enforcement is described in more detail in section 6.2.

6.1.1. Object Model

All user data is managed by the module as objects. Objects are owned by external processes/users and manipulated by the module. They are characterized by different attributes used by the module to determine the handling rules to be applied. The module provides two ways of storing objects: permanent (also known as PKCS #11 token objects) and volatile (also known as session objects). Permanent objects are kept inside the module even when no power is applied to it. They are stored encrypted in a flash memory device. Session objects only exist when power is applied to the module and they are stored in volatile RAM.

The Luna object model is very closely related to the PKCS#11 standard. More details on the Luna interface exists in the Luna Interface Control Document [8].

6.1.2. Multi-Session Capability

The Luna PCI cryptographic module manages communication with external processes on a per session basis. Applications running on the host system requiring data and cryptographic services from the module have to open a session with the module before gaining access to the module's functions and objects. The session provides a logical connection between the application and the module and it is the session to which the authentication state is bound. It is possible for an application to open multiple sessions with the module or have multiple applications each opening various sessions with the module.

The module provides a higher level of connection abstraction based on an access id that associates a group of sessions to a particular application. This approach allows an application or applications to share sessions, and associated authentication state, within the scope of that access id.

6.1.3. TOE Roles

The following roles are supported by the TOE:

- Security Officer (SO) – authorized to install and configure the TOE, set and maintain security policies, and create and delete users (Crypto Officer and Crypto User roles). The TOE can have only one SO.
- Crypto Officer – authorized to create, use, destroy and backup/restore cryptographic objects.
- Crypto User – authorized to use cryptographic objects (e.g., sign, encrypt/decrypt).

The Crypto Officer and Crypto User interface to the Luna PCI for cryptographic operations using the PKCS #11 API. The Security Officer uses a separate Command Line Interface (CLI), which is part of the interface software, to perform configuration, security policy settings and user creation/deletion. The CLI is also used by the Crypto Officer to perform backup and restoration of cryptographic objects.

The TOE allows for the creation of multiple users in the Crypto Officer and Crypto User roles. Each user is created within a cryptographically separated partition in the Luna PCI cryptographic module and each partition must have one and only one user in the Crypto Officer role. A partition may also have one and only one user in the Crypto User role. It is possible to have up to twenty (20) partitions defined within the Luna PCI cryptographic module.

6.1.4. Multi-User Capability

A user must access the module through a session. Sessions are opened as Public sessions and may remain Public or become Private (authenticated) following a successful user authentication. Session states are kept separate based on the user authentication state ensuring that sessions cannot be shared among users. The module allows multiple user identities to be authenticated at a time. Once authenticated, a session becomes bound to the user identity and has access to all cryptographic operations appropriate to the user's role and may access private objects generated on behalf of the user in previous sessions. Although there may be many users authenticated to the cryptographic module, there is effectively only one thread of execution within the module and, therefore, only one command being executed from request through to response at any given time.

6.2. Capability and Policy Settings

The Luna PCI was designed with the flexibility needed to support a number of different product variants. The main method used to control the behaviour of different products is a fixed set of "capabilities" set at the factory. The settings made for the TOE configuration are shown in sub-sections 6.2.1 and 6.2.2. For each of the capabilities, a corresponding policy element exists. The SO establishes the policy that will govern the cryptographic module's operation, according to the requirements of the customer organization, by enabling/disabling or refining the corresponding policy elements to equate to or to be more restrictive than the pre-assigned capabilities. See section 6.3.14 for a description of the specific policy elements that are configurable by the SO in the TOE configuration.

Policy set elements can only refine capability set elements to more restrictive values. Specifically, if a capability is set to allow, the corresponding policy element may be set to either enable or disable. However, if a capability is set to disallow, the corresponding policy element is set to disabled and is not SO-configurable. Thus, an SO cannot use policy configuration to lift a restriction set in a capability definition.

There are also several elements of the cryptographic module's behaviour that are truly fixed for all product variants and, therefore, are never subject to configuration by the SO. The specific elements are the following:

- Non-sensitive secret keys are not allowed.
- Non-sensitive private keys are not allowed.
- Non-private (Public) secret keys are not allowed.
- Non-private (Public) private keys are not allowed.
- Creation of secret keys and private keys through the PKCS #11 create object interface is not allowed. That is, the API cannot be used to create keys by passing in known plaintext values.

In the next two sub-sections, all capability elements described as "allow/disallow some functionality" are Boolean values where false (or zero) equates to disallow the functionality and true (or one) equates to allow the functionality. Except as noted, all Boolean capabilities are Allowed, thus leaving them configurable by the SO. The remainder of the elements are integer values with either the default value or the maximum in number of bits shown.

6.2.1. HSM Level Capabilities

The following is the set of capabilities supported at the HSM level:

- Allow/disallow non-FIPS algorithms available.
- Allow/disallow password authentication (disallowed in TOE configuration).
- Allow/disallow Remote PED Usage (disallowed in TOE configuration).
- Allow/disallow M of N.
- Allow/disallow cloning.
- Allow/disallow masking (disallowed in TOE configuration).
- Allow/disallow M of N auto-activation.
- Allow/disallow ECC mechanisms.
- Allow/disallow Remote Authentication.
- Allow/disallow SO reset of partition PIN.
- Allow/disallow network replication.
- Allow/disallow forcing change of User authentication data.
- Number of failed SO logins allowed before the HSM is zeroized (set to 3, non-configurable).

Note that the module policy “Allow/disallow Remote PED Usage” is disallowed with the TOE in the evaluated configuration. The TOE must be purchased in the “Local PED” configuration. Use of a Remote PED, which requires the module policy “Allow/disallow Remote PED Usage” to be changed to “Allow” means that the TOE can no longer be considered an evaluated configuration.

6.2.2. Partition Level Capabilities

The following is the set of capabilities supported at the partition level.

- Allow/disallow partition reset.
- Allow/disallow activation.
- Allow/disallow automatic activation.
- Allow/disallow High Availability.
- Allow/disallow multipurpose keys.
- Allow/disallow changing of certain key attributes once a key has been created.
- Allow/disallow operation without RSA blinding.
- Allow/disallow signing operations with non-local keys.
- Allow/disallow raw RSA operations.
- Allow/disallow private key wrapping (disallowed in TOE configuration).
- Allow/disallow private key unwrapping.
- Allow/disallow secret key wrapping
- Allow/disallow secret key unwrapping.
- Allow/disallow Level 3 operation without a challenge (disallowed in TOE configuration).
- Allow/disallow user key management capability. (Allowed in TOE configuration. This would be disabled by the SO at the policy level to prevent any key management activity in the partition, even by a user in the Crypto Officer role. This could be used, for example, at a CA once the root signing key pair has been generated and backed up, if appropriate, to lock down the partition for signing use only.)
- Allow/disallow incrementing of failed login attempt counter on failed challenge response validation.

- Allow/disallow RSA signing without confirmation.
- Allow/disallow RA type wrapping (disallowed in TOE configuration).
- Minimum/maximum password length (not applicable in TOE configuration).
- Level of storage space available for key storage (4 bits).
- Number of failed Partition User logins allowed before partition is locked out/cleared (default is 10, SO can configure it to be 3 <= N <= 10)

The following capabilities are only configurable if cloning is allowed and enabled at the cryptographic module level:

- Allow/disallow private key cloning (allowed in TOE configuration).
- Allow/disallow secret key cloning (allowed in TOE configuration).

The following capabilities are only configurable if masking is allowed and enabled at the cryptographic module level:

- Allow/disallow private key masking (disallowed in TOE configuration).
- Allow/disallow secret key masking (disallowed in TOE configuration).

6.3. IT Security Functions

6.3.1. Audit Data Generation

The TOE provides raw audit data that can be read and interpreted by an audit application on the host computer system. The audit data consists of the session handle, command code and input parameters transmitted to the TOE, the output parameters in response to the command and the error return code (0 if the command was successful). The session handle associates the user to the command code. Each command code and corresponding response has a sequence number associated with it. This allows the audit application to maintain sequence ordering and to unambiguously associate responses to commands. The following events are covered by the audit data:

- TOE initialisation
- TOE start-up
- Generation of secret key and asymmetric key pair
- Destruction of key material
- Use of secret key and private key
- Unsuccessful authentication
- Modification of TOE management data
- Adding new users
- Deleting users
- Execution of the TSF self tests during initial start-up and at the request of the authorised user
- Unsuccessful self test operations
- Execution of the cloning functions for key backup and restore operations

The audit records are written by the TSF to a circular storage buffer that can be read but not written by an audit administrative application. The buffer ensures the availability of approximately 60 kB of stored records.

6.3.2. Trusted Path – Luna PED

User authentication data and other critical security parameters are protected through the use of a separate port and data path for their transfer, and by providing mechanisms to protect their confidentiality and integrity. Attached to this separate data port is the Luna PIN Entry Device or Luna PED.

The Luna PED, with accompanying iKeys, is depicted in Figure 4. It houses a number of input/output interfaces that, in combination, provide a trusted path device for the communication of authentication data and critical security parameters to and from the Luna PCI cryptographic module. The Luna PED has a character display used to display status and prompt messages, and a challenge secret that is output by the cryptographic module at the time a partition is created [see sub-section 6.1.3]. It has a keypad used to enter simple responses (Yes/No/Enter) and to enter an optional PIN that is combined with the authentication data stored on an iKey as part of the authentication process. It has a USB receptacle for the input/output of data to the iKey and it has a serial communications cable that connects to the separate data port, which is wired directly to the cryptographic module. Because the PED has a direct serial communications interface to the cryptographic module, only local entry of iKey authentication data is possible.

The following types of iKey are used with the Luna PED:

- Blue (SO) iKey – for the storage of SO authentication data,
- Black (User) iKey – for the storage of User authentication data,
- Red (Domain) iKey – for the storage of the cloning domain data, used to control the ability to clone from a cryptographic module to a backup token,
- Green (M of N) iKeys – used to store M of N secret shares, used for multi-purpose control of critical functions,

Any iKey, once data has been written to it, is an Identification and Authentication device and must be safeguarded accordingly by the administrative or operations staff responsible for the operation of the TOE within the customer's environment.

Figure 5 Luna® PED with iKeys

6.3.3. User Identification and Authentication

The iKey contains the user's identification number and the pseudo-randomly generated 48-byte authentication secret for the user and is entered into the key receptacle in the PED in order to identify and authenticate the user.

A user is defined as an entity that acts to perform an operation on the TOE. In most instances, this will be a host application program such as a PKI Certification Authority implementation. The TOE supports three user roles; Security Officer (SO), Crypto Officer and Crypto User. For a user to assume any role the module enforces user identification and authentication.

The TOE requires that all users (SO, Crypto Officer and Crypto User roles) be authenticated by proving knowledge of a secret shared by the user and the cryptographic module.

The TOE generates the authentication secrets using its PRNG. For the SO, the authentication secret is a 48-byte random secret and it is generated at the time the cryptographic module is initialised. For Users, the authentication secrets consist of a 48-byte random secret and separate challenge secret(s); these are generated at the time the partition is created by the SO. The authentication secret(s) are provided to the operator via the Luna PED display and iKey, as described in sub-section 6.3.2, and must be entered by the operator via the Luna PED and via a logically separate trusted channel (in the case of the response based on the challenge secret) during the login process. Both the Crypto Officer and Crypto User use the same 48-byte random secret. If a Partition is created with Crypto Officer and Crypto User roles, a separate challenge secret is generated for each role.

SO authentication requires the transmission to the cryptographic module of the Blue iKey data combined with the optional PIN through the trusted path.

User authentication is a two-stage process. The first stage is termed “Activation” and is performed using the Luna PED. Activation requires the transmission to the cryptographic module of the Black iKey data combined with the optional PIN through the trusted path. Once Activation has been performed, the partition data is ready for use within the cryptographic module. Access to key material and cryptographic services, however, is not allowed until the second stage of authentication, equivalent to “User Login”, has been performed. This typically requires the input of a partition’s challenge secret as part of an application program’s login operation.

The authentication challenge secret (or secrets if the Crypto Officer and Crypto User roles are used) for the partition is generated by the cryptographic module as a random 75-bit value that is displayed as a 16-character string on the visual display of the trusted path device. The challenge secret is then provided, via a secure out-of-band means, to each external entity authorized to connect to the partition and is used by the external entity to form the response to a random one-time challenge from the cryptographic module. The encrypted one-time response is returned to the cryptographic module where it is verified to confirm the “User Login”.

Following a successful login, the user is bound to the subject acting on its behalf by having the User Authorization Vector (UAV) data included in the state data maintained by the session manager. In the case of the Luna PCI cryptographic module the subject acting on behalf of a user is a session. The relationship between the user and the session is discussed in more detail in section 6.1.4 and the data contained in the UAV is described in section 6.3.5.

The TOE also enforces a maximum login attempts policy. This feature serves to prevent an exhaustive search approach to find the authentication data of the SO or a User. The implementation of this feature differs for an SO authentication data search and a User authentication data search.

In the case of a user:

> If “y” consecutive user logon attempts fail (“y” is defined by the SO in the configurable policy for the partition), the TOE will either lock the partition or erase the partition, as defined by the SO in the configurable policy. If it has been locked, the partition must be unlocked by the SO in order to allow user login. If it has been erased, the partition cannot be recovered directly. If recovery is required, the SO must create a new partition and the new Crypto Officer must recover the partition’s data from a backup token.

In the case of the SO, if three (3) consecutive SO logon attempts fail, the module is zeroized and must be re-initialized.

6.3.3.1. M of N Activation

The TOE can also be configured by the SO to require the use of an M of N secret sharing authentication scheme to enable the module for operation. The M of N activation protocol provides the capability to enforce multi-person integrity over SO operations and activation of each partition.

The M of N capability is based on Shamir’s threshold scheme. The Luna PCI cryptographic module generates a 32 byte secret and protects it by “splitting” it into “N” pieces and storing each piece on an iKey dedicated to that purpose (Green Key). Any “M” of these “N” pieces must be transmitted to the Luna PCI cryptographic module by inserting the corresponding iKeys into the Luna PED in order to reconstruct the original secret.

6.3.3.2. Unidentified and Unauthenticated Users

The TOE allows the following actions on behalf of the user to be performed before the user is identified:

- Perform start-up, self-test (FPT_TST.1), detection of the secure blocking state (FPT_FLS.1), detection of violation of physical integrity (FPT_PHP.2),
- Perform basic diagnostic functions, such as checking the communications from the host to the card, checking firmware level and token info and checking information on mechanisms supported.

The user must be identified before any other TSF-mediated action is allowed to proceed.

The TOE allows the following actions on behalf of the user to be performed before the user is authenticated:

- Perform start-up, self-test (FPT_TST.1), detection of the secure blocking state (FPT_FLS.1), detection of violation of physical integrity (FPT_PHP.2),
- Perform basic diagnostic functions, such as checking the communications from the host to the card, checking firmware level and token info and checking information on mechanisms supported.
- Open a session
- Access Public data objects
- Identification (FIA_UID.1).

The user must be authenticated before any other TSF-mediated action is allowed to proceed.

6.3.4. Authentication Data Selection

The User authentication data is a 48 byte value that is randomly generated by the module and stored on an iKey (Blue for SO or Black for User) plus the Crypto Officer and Crypto User Challenge Secrets, which are initially provided to the user via the PED display. The iKey represents the user to the module and, therefore, must be properly protected within the environment in which the module operates. A User, in the Crypto Officer role, and the SO can request to change their respective authentication data at any time using the Command Line Interface.

6.3.5. User Account Data

The Security Officer is the only role allowed to create users, modify user status and delete users. The TOE maintains a user's account data in a User Authorization Vector (UAV) that is stored in memory reserved for the TOE's use. The UAV includes the following data:

- User ID number
- User checkword
- User function vector
- User failed login count
- User "lockout" status

The User checkword contains the User's secret key, Crypto Officer and Crypto User Challenge Secrets, and a validation string encrypted using a key derived from the User's authentication data. The secret key is randomly generated by the module at the time the User is created and is used to encrypt a User's objects on the module. The validation string is a known byte string used to verify that the checkword has been decrypted correctly.

6.3.6. Access Control

The TOE enforces an identity-based access control policy that applies to all objects on the module, in particular to private key and secret key objects, and governs a subject's access to an object using the following operations:

- Read (Query Attribute Value)
- Modify
- Destroy
- Generate[27]
- Wrap (export)
- Use[28]
- Clone

A subject's access to objects stored on the module is mediated on the basis of the following subject and object attributes:

- Subject attributes:
    - Session and Access ID
    - User ID associated with session (Access Owner)
    - Role.
- Object attributes:
    - Private. If True, object is Private. If False, object is Public.
    - Owner. Object ownership is assigned to the object creator, if the object is Private. Public objects are not owned by a user. Ownership is enforced by user identity and internal key management.
    - Sensitive. If True, object is Sensitive. If False, object is Non-Sensitive.
    - Extractable. If True, object may be extracted. If False, object may not be extracted.
    - Modifiable. If True, object may be modified. If False, object may not be modified.

Private data objects are labelled with a number corresponding to their owner and sensitive attributes are encrypted using the owner's secret key. Private data objects are only accessible by the object owner. Public data objects may be accessed by any user with an active session on the module. Secret key and private key objects are always created as Private, Sensitive objects and can only be used for cryptographic operations by a logged in User. Only data and certificate objects can be non-sensitive. Secret key objects that are marked as extractable may be exported from the module using the Wrap operation. Private keys are never extractable from the Luna PCI cryptographic module.

The module does not allow any granularity of access other than owner or public (i.e., a Private data object cannot be accessible by two users and restricted to other users). Ownership of an object gives the owner access to the object through the allowed operations but does not allow the owner to assign a subset of rights to other users. Allowed operations are those permitted by the configurable policy settings and the access matrix in section 5.1.3.4.

[27] The Generate operation is intended primarily to indicate symmetric key or asymmetric key pair generation. However, it also includes other methods of creating an object in the TOE, such as importing (unwrapping) a key and generic data object creation.
[28] The Use operation includes symmetric key encryption/decryption, private key signing and decryption, and public key verification and encryption.

6.3.7. Object Reuse

The TOE enforces an object reuse policy in that every object is allocated its own portion of memory (flash or volatile RAM). Permanent objects (stored in flash) are maintained in an encrypted state at all times, and their information content is, therefore, never available except when decrypted for use in volatile memory within the TSF boundary. The policy also ensures that no permanent object is placed in a previously allocated memory location unless all previous memory content is purged and zeroized. When cryptographic functions are performed, a cryptographic context is created to hold data required by the function (e.g., a DES key schedule for a DES function). The cryptographic context only exists in volatile RAM memory and is not accessible to any functions except those defined by its owner function. The memory assigned to a cryptographic context is always purged of its content before being handed over to another function. Direct access to either volatile or flash memory locations is never provided to users; all user interaction with the objects within the module is via memory handles.

6.3.8. Data Authentication

The TOE provides data authentication at two different levels. At the first level, the TOE calculates the SHA-1 fingerprint of each object it stores and the user may query the value of the fingerprint at any time. This allows the user to verify the continuing validity of the object.

At the second level, the TOE will generate evidence of the validity of a private key and its corresponding public key in a special digitally signed certificate format, known as a Public Key Confirmation. The signature is performed using a private key that is either generated by SafeNet specifically for this purpose and whose public key certificate has been signed by the SafeNet trust anchor or generated by a customer organization and whose public key certificate has been signed by a third-party CSP or Trust Centre. The Public Key Confirmation permits a user to verify the validity of an asymmetric key pair, verify that the TOE generated it and identify the trusted third party providing the guarantee of validity and origin.

6.3.9. Key Pair Integrity Checking

The TOE provides a function to verify the integrity of an asymmetric key pair before outputting a digital signature performed using the private key from the pair. The integrity of the private key is checked before output of a digital signature by verifying the digital signature created with the private key using the corresponding public key object for signature verification. If the verification fails, the signature is not output and an error code only is returned.

6.3.10. Key Export and Import Protection

Secret keys may only be exported from the TOE boundary in a wrapped (encrypted) form if the Extractable attribute is True. Private keys may never be exported from the TOE boundary. Secret keys are exported from the module without their associated security attributes. If the Extractable attribute is False, the key may not be exported from the module boundary under any condition.

Objects may be imported into the module under the control of the Access Control policy. Secret keys and/or private keys generated in the host IT environment may only be imported into the module by an unwrapping operation on the module. Any attributes of keys imported in this way are ignored by the TOE and their attributes are set to default values by the TOE. Unwrapped keys have their Sensitive attribute set to True by the TOE. The configurable policy for a partition may also be set to prohibit the use of externally generated private keys for signing operations.

Wrapping and unwrapping of key material between the TOE and other entities can only take place if prior agreement has been reached regarding the key to be used for the wrap and unwrap operations. This can either be through key sharing of a secret key for use with a symmetric encryption algorithm or through the use of the public key of the intended recipient with an asymmetric encryption algorithm.

6.3.11. Cryptographic Material Management

Cryptographic material (key) management functions protect the confidentiality of key material throughout its life-cycle. The key management functions provided by the TOE are the following:

- Cryptographic key generation in accordance with the following indicated standards:
  - RSA 1024, 2048, 4096 bits key pairs in accordance with ANSI X9.31.
  - TDES 112, 168 bits in accordance with FIPS PUB 46-3 and ANSI X9.52.
  - AES 128, 192, 256 bits in accordance with FIPS PUB 197
  - DSA 1024 bits key pairs in accordance with FIPS PUB 186-2.
  - ECDSA in accordance with FIPS PUB 186-2 and ANSI X9.62.
- Secure key access following the PKCS #11 standard.
- Destruction of cryptographic keys in accordance with the FIPS PUB 140-2 Level 3 standard.

An object on the module that is destroyed using the PKCS #11 function C_DestroyObject (the user delete command available through the API) is marked invalid and remains encrypted with the user's secret key until such time as its flash locations are re-allocated for additional data on the module; at which time they are purged and zeroized before re-allocation. The same strategy of marking an object invalid and purging the memory content before re-allocation is followed for volatile memory as well as flash.

Objects on the module that are destroyed as a result of authentication failure are zeroized (all flash blocks in user's memory turned to 1's). If it is an SO authentication failure all flash blocks on the module are zeroized.

Objects on the module that are destroyed through C_InitToken (the SO function to initialize the module available through the API) are zeroized, along with the rest of the flash memory being used by the SO and User.

All cryptographic material management functions are performed in the module in accordance with the appropriate cryptographic standards using algorithms and mechanisms that have been formally validated as meeting the FIPS PUB 140-2 Level 3 standard.

6.3.11.1. Key Storage and Access Protection

Keys are always stored as secret key or private key objects with the Sensitive attribute set and, therefore, with the key value encrypted. Access to keys is never provided directly to a calling application. A handle to a particular key is returned that can be used by the application in subsequent calls to perform cryptographic operations. Key storage and access is performed in accordance with the PKCS #11 object model and function specifications.

6.3.12. Cryptography

Because of its generic nature, the Luna PCI cryptographic module firmware supports a wide range of cryptographic algorithms and mechanisms, a number of which are not relevant to the CMCSOB PP specifically. The cryptographic functions and algorithms that are relevant to the TOE are the following:

- Random Number Generation
  - FIPS 140-2 validated Deterministic Random Bit Generator (Pseudo-random Number Generator) seeded by internal Hardware Non-deterministic Random Bit Generator
  - Based on ANSI X9.31, Appendix A section 2.4
- Compute Digital Signatures And Verify Digital Signatures
  - RSA 1024 bits, 2048 bits, 4096 bits (PKCS #1 V1.5, PKCS #1 PSS, ANSI X9.31) with SHA-1
  - RSA 1024 bits, 2048 bits, 4096 bits (PKCS #1 V1.5, PKCS #1 PSS) with SHA-256, 384, 512

  - DSA 1024 bits (FIPS PUB 186-2) with SHA-1
  - ECDSA (FIPS PUB 186-2 Appendix 6 recommended curves) with SHA-1
- Encrypt / Decrypt Data
  - RSA 1024, 2048 and 4096 bits in accordance with PKCS #1 V1.5 and OAEP
  - TDES (ECB and CBC mode) 112 and 168 bits in accordance with FIPS PUB 46-3
  - AES (ECB and CBC mode) 128 and 256 bits in accordance with FIPS PUB 197
- Export (Wrap) and Import (Unwrap) Secret Keys
  - TDES, AES with TDES and AES in ECB mode
  - TDES, AES with RSA 1024, 2048 and 4096 bits in accordance with PKCS #1 V1.5

The necessary keying material needed by these algorithms may be generated or derived on-board. Random data needed to produce sound key material is generated by the module's PRNG. In some cases, key material may be imported from an external source in an encrypted (wrapped) form and decrypted (unwrapped) inside the module.

6.3.13. Data Exchange

The TOE provides security functions that support secure data exchange in two main ways:

- Data integrity and authenticity is protected through the use of RSA and DSA digital signatures. The digital signature of the data object provides evidence of data validity. The TOE provides logged in Users the ability to generate evidence in the form of a digital signature provided they have access to the private signing key and to verify the evidence and the identity of the originator who generated the evidence provided they have possession of the digitally signed information and access to the signer's verification public key.
- Data confidentiality is protected through the use of symmetric and/or asymmetric encryption/decryption of user data and in the Wrapping and Unwrapping operations.

6.3.14. Specification of Security Management Functions

The TOE provides the following security management functions:

- disable, enable and modify the behaviour of configurable policy settings at the HSM and Partition levels (FMT_MOF.1),
- change_default, query, modify and delete the security attributes User Locked Flag,
- modify the security attributes UAV – Checkword,
- change_default and delete the security attributes User ID and UAV – Checkword,
- change_default and modify the security attributes SOV – Checkword,
- modify the security attributes CKA_PRIVATE (for data and certificate objects only), CKA_EXTRACTABLE (for secret keys only), CKA_DERIVE (for secret keys only) and CKA_MODIFIABLE,
- change_default the Number of User Login Failures Allowed.

Details of these management capabilities are provided in sections 6.3.15 and 6.3.16.

6.3.15. Security Function Management

The TOE provides security management capabilities for the Security Officer (SO) to disable, enable and modify the behaviour of the functions listed below.

The following is the set of policies supported at the HSM level:

- Enable/disable non-FIPS algorithms available.
- Enable/disable trusted path authentication (allowed and must be enabled in TOE configuration).
- Enable/disable M of N.
- Enable/disable cloning.
- Enable/disable M of N auto-activation.
- Enable/disable ECC mechanisms.
- Enable/disable Remote Authentication.
- Enable/disable SO reset of partition PIN.
- Enable/disable network replication.
- Enable/disable forcing change of User authentication data.
- Enable/disable Remote PED Usage (disallowed in TOE configuration).

Note that the module policy “Allow/disallow Remote PED Usage” is disallowed with the TOE in the evaluated configuration. The TOE must be purchased in the “Local PED” configuration. Use of a Remote PED, which requires the module policy “Allow/disallow Remote PED Usage” to be changed to “Allow” means that the TOE can no longer be considered an evaluated configuration.

The following is the set of policies supported at the partition level:

- Enable/disable partition reset.
- Enable/disable activation.
- Enable/disable automatic activation.
- Enable/disable High Availability.
- Enable/disable multipurpose keys.
- Enable/disable changing of certain key attributes once a key has been created.
- Enable/disable operation without RSA blinding.
- Enable/disable signing operations with non-local keys.
- Enable/disable raw RSA operations.
- Enable/disable private key unwrapping.
- Enable/disable secret key wrapping
- Enable/disable secret key unwrapping.
- Enable/disable user key management capability. (This would be disabled by the SO at the policy level to prevent any key management activity in the partition, even by a user in the Crypto Officer role. This could be used, for example, at a CA once the root signing key pair has been generated and backed up, if appropriate, to lock down the partition for signing use only.)
- Enable/disable incrementing of failed login attempt counter on failed challenge response validation.
- Enable/disable RSA signing without confirmation.
- Enable/disable private key cloning.
- Enable/disable secret key cloning.

6.3.16. Security Data Management

The TOE allows the Security Officer and the Crypto Officer to manipulate security-relevant data stored on the module. Specifically, it allows only the Security Officer to change the default values of the settings listed below:

- Number of failed Partition User logins allowed before partition is locked out/cleared. (Default is 10, SO can configure to be 3 <= N <= 10)

The User Authorization Vector, described in section 6.3.5, is the data structure used by the module to store the user's security attributes. The TOE restricts the ability to manipulate the UAV data as described below:

- Only the Security Officer role can change_default, query, modify and delete the UserLockedFlag.
- Only the Security Officer role can change_default and delete the:
    - UserID.
    - Checkword, which includes the user secret key plus a fixed value used for authentication in encrypted form.
- Only the Security Officer and User roles can modify the Checkword (for the SO or applicable User ID).

The Token Access Control policy also restricts the ability to modify, the security attributes CKA_PRIVATE (for data and certificate objects only), CKA_EXTRACTABLE (for secret keys only), CKA_DERIVE (for secret keys only) and CKA_MODIFIABLE to the Crypto Officer role.

The TOE assigns default attributes to objects as they are created. The creator of the object may specify values different from the defaults with the exceptions described below.

There are security-relevant object attributes that are set to restrictive default values that cannot be changed by anyone. These attributes and their settings are the following:

- The CKA_SENSITIVE attribute is set TRUE for all secret and private key objects.
- The CKA_EXTRACTABLE attribute is set FALSE for all private key objects.

The TOE restricts the ability to query the audit data to the Crypto Officer role. The SFR (5.1.5.9) also refers to the Auditor role; this role is, however, held by the TSF.

6.3.17. Logical Self-Protection of Security Functions

The TOE ensures the logical protection of its security functions from attempts to subvert or bypass security enforcement by implementing a number of self-protection measures. The main self-protection features are described below.

6.3.17.1. Memory and Firmware Integrity Check

The firmware integrity is protected by an error detection code based on a Cyclic Redundancy Check (CRC) and a cryptographic hash function. The firmware's integrity is checked by the bootblock using the CRC when the firmware is initially loaded or updated and every time the module is started. The firmware also verifies the SHA-1 hash of the loaded firmware before it starts executing. The module will halt if the firmware integrity is not verified. Similarly, the module's memory is checked for consistency every time the module is started and the module will halt if the memory consistency check fails.

6.3.17.2. Self-Tests

The TOE performs a number of tests of security-critical functions each time it is activated and on demand from a user. The TOE offers three categories of self-tests that can be called up by the user at any time: Hardware, cryptographic and PRNG checks. The hardware self test verifies access to all of the volatile RAM memory. The cryptographic self-tests perform a test of all of the cryptographic algorithms provided by the module. The cryptographic and PRNG self-tests are based on a known answer test methodology where a known key, or initial

configuration, is used to process a known data input and the result obtained is compared to a previously-calculated answer.

6.3.17.3. Prevention of By-pass and Separate Execution Domain

The TOE prevents bypass by ensuring that TSP enforcement functions are invoked and succeed before allowing a subsequent firmware function to proceed. It maintains a separate domain for its own execution that is protected from external agents. It also separates users by encrypting private objects with the user's secret key and by allowing only one thread of execution on the module at any one time and, therefore, allowing only one user's command to be active at any time.

6.3.17.4. Preservation of Secure State

The TOE preserves itself in a secure state in the event of failures detected by the abstract machine test and self-test functions. Behaviour in the event of other failure conditions is described in sub-section 6.3.20.

6.3.17.5. Firmware Loading and Firmware Update

The Luna® PCI cryptographic module requires the use of a cryptographically protected trusted channel for initial firmware loading at the factory prior to delivery to the customer and when the firmware is later updated at the customer's site. The trusted channel is provided as described in the following two paragraphs.

Firmware can only be initially loaded onto Luna® PCI cryptographic module from a separate module dedicated for the purpose and containing a firmware image that has been digitally signed by SafeNet and encrypted using a secret key generated specifically for this purpose and if the module itself is a valid Luna® PCI cryptographic module.

For firmware updates, the updated image is signed and encrypted using a dedicated module at SafeNet and distributed to customer sites in software form along with a separately distributed authorization code. The TOE verifies the digital signature to ensure that the updated image originated at SafeNet and that it has not been modified. The TOE decrypts the image using a key derived from the authorization code to ensure that its confidentiality has been protected while in transit. The use of the authorization code ensures that only authorized customers may perform the update and ensures that only valid Luna® PCI cryptographic modules can decrypt the image in order to perform the update. The trusted channel for communicating the firmware image from the dedicated SafeNet module to the target Luna® PCI cryptographic module is initiated by the dedicated module because it is the one that generates the symmetric keys and authorization code, digitally signs the image and encrypts it for transmission to the target. The actual firmware update process is performed by the target module and the first part of that process completes the communication that was initiated by the dedicated SafeNet module by verifying the digital signature of the image and decrypting it for loading into the TOE.

6.3.18. Cloning

For performance and secure backup purposes, Luna PCI cryptographic modules and Luna backup tokens may be grouped in clusters that are referred to as "domains." A domain is established by generating a 24 byte secret, known as a cloning domain key or cloning domain identifier, on one module (that could be considered to be the "master" for the domain) and transferring the secret securely via the PED to other modules or backup tokens that are to be part of the domain. The cloning domain key is then used during the mutual authentication and key agreement exchange that takes place between modules, or between a Luna PCI and a backup token, as described briefly below. This mutual authentication ensures that the two modules participating in the cloning operation belong to the same cloning domain and can thus participate in the cloning process.

When modules are members of a domain, they must be capable of operating in such a way that they behave as one identical module to the calling application. The cloning function provides the capability to duplicate the cryptographic state of a module by cloning token objects from a source module to a target module within the same cryptographic domain in a cryptographically protected fashion that prevents modification and disclosure.

When cloning is invoked, the cloning protocol protects security-relevant data from disclosure and modification when it is transferred between the TOE and the remote trusted component (backup token or another Luna PCI module). The protocol is designed such that source and target modules both participate in ensuring that objects are all transferred correctly between modules. It also ensures that any data exchanged during the cloning operation cannot be replayed in order to gain unauthorized access to the module. The source module maintains its original state and, therefore, any sort of failure of the cloning function will not result in a loss of use of the original objects.

The cloning protocol implements a mutual authentication mechanism to ensure that both modules are members of the same domain by providing mutual authentication of the two modules. The mechanism uses cryptographic techniques to provide mutual authentication, proof of origin, integrity and confidentiality of the objects being transferred from source to target module within a domain. The key management scheme used within the cloning protocol also protects against replay attacks and minimizes the impact of possible key compromise by ensuring that a unique TDES key is used for each cloning operation.

6.3.19. Physical Self-Protection

Tamper-evident features are implemented in the manufacture of the module. Any tampering that might compromise the module's security is detectable by visual inspection of the physical integrity of the module. The enclosure covers are bonded to the circuit card assembly and an attempt to remove either of the covers will result in significant damage to the card, rendering the module inoperable.

The module's physical design resists visual inspection of the device design, physical probing of the device and attempts to access sensitive data on individual components of the device and provides evidence of the occurrence of such physical tampering.

The module responds automatically to attempts to open its enclosure by ensuring that plaintext key material and other sensitive data is erased from the module.

6.3.20. Failure Handling

If power is lost to the module for whatever reason, permanent objects (private keys, etc.) are preserved and remain cryptographically protected; session objects are cleared from the module. The module can be placed back into operation without compromise of its functionality or permanently stored data. In case of power failure in the host IT environment, host system restart or other circumstances that do not affect the module's operational capability, the module will ensure continued protection of sensitive material and will permit recovery from the last logged in state.

Data input/output failures would only affect the processing of the current command and, because no PKCS #11 API function returns sensitive plaintext data, there could be no compromise of the user data protection capabilities. Because of the way in which commands are handled, the module would remain in the state it was at the last successful command completion. When data input/output capability is restored the module would resume operation in that state.

6.3.21. Backup and Recovery

As described in sub-sections 6.3.17.4 and 6.3.20, the module maintains its secure state in the event of a failure. Depending on the nature of the failure, the module will maintain its secure state and resume operation as described below:

- In the event of host system discontinuity the module maintains its current logged in state and resumes that state when the host system restarts.
- In the event that power is lost to the module for a short time (less than 1.5 hours), it can maintain its authentication state using the Auto-activation feature. This allows it to resume operation after power has been restored without requiring the User to activate the partition using the PED and iKey. It will resume

operation with all security properties intact but the operational state of the module prior to loss of power will be lost.

- In the event that power is lost to the module for a longer period of time, it maintains its secure state by maintaining the encryption of all sensitive data and it requires the User to activate the partition prior to resuming operation. It will resume operation with all security properties intact but the operational state of the module prior to loss of power will be lost.
- In the event of a catastrophic damage to or failure of the module itself, recovery is accomplished by inserting and activating a backup module, as described below.

The TOE provides the capability to securely backup a module using the cloning function. Because the cloning function securely duplicates all objects from the primary module to the backup token, the backup token allows recovery from the backup token by cloning the backed up objects to a new module that has been initialized with the same cloning domain. The basic data authentication mechanism described in section 6.3.8 can be used at both the TOE and the backup token before and after cloning operations to ensure the integrity of backed up and restored key objects.

6.4. Strength of Function

The minimum Strength of Function (SoF) required for the TOE is **SoF - high**. This applies to the following security functions:

- Cryptography – Random Number Generation, section 6.3.12
- User Identification and Authentication, section 6.3.3
- Authentication Data Selection, section 6.3.4
- Cloning, section 6.3.18 (The cloning domain identifier, generated and used as part of the cloning transfer, authenticates the source and target modules as part of the same cloning domain and, therefore, trusted to communicate.)

The above listed functions are those that utilize probabilistic or permutational mechanisms.

The TOE provides this Strength of Function based on the following:

- Authentication requires a protected data storage device to hold one component of the authentication data, a specially constructed PIN Entry Device and the use of designated PCI Card pins in order to input authentication data to the module.
- The authentication data stored on the protected data storage device is 48 bytes in length and is generated by the TOE's pseudo-Random Number Generator (RNG), which itself has a rating of SoF-high.
- The M of N secret sharing scheme that may also be employed for authentication is based on a recognized strong secret sharing algorithm, with the base secret being 256 bits in length.
- The cloning domain identifier used to authenticate source and target modules as part of the cloning protocol is generated using the TOE's PRNG and is 168 bits in length.

The Strength of Function (SoF) claimed for the Pseudo-Random Number Generator (PRNG) used by the authentication mechanism is SoF-high based on its having been validated as part of the Luna PCI validation against the FIPS 140-2 standard for cryptographic modules, and on evidence provided and testing for conformance to the standard established by the scheme.

6.5. Assurance Measures

The assurance requirements for this TOE are as specified in the EAL 4 package augmented by:

- ADV_IMP.2 (Implementation of the TSF)
- ALC_FLR.2 (Flaw Reporting Procedures)
- AVA_CCA.1 (Covert channel analysis)
- AVA_MSU.3 (Analysis and testing for insecure states)
- AVA_VLA.4 (Highly resistant)

Evidence, in the form of documentation, plans and procedures that meet the content and presentation requirements of Part 3 of the Common Criteria, is provided to satisfy the specified assurance requirements. References to the appropriate supporting documentation are provided in Table 8-9 – Assurance Measures.

The evidence includes deliverables in the following categories:

1. Security Target
2. ST Rationale
3. Functional Specification (ADV_FSP.2)
4. High Level Design (ADV_HLD.2)
5. Low Level Design (ADV_LLD.1)
6. Implementation Representation (ADV_IMP.2)
7. Representation Correspondence (ADV_RCR.1)
8. Security Policy Model (ADV_SPM.1)
9. Developer's Tests (ATE_COV.2, ATE_DPT.1, ATE_FUN.1, ATE_IND.2)
10. Configuration Management (ACM_AUT.1, ACM_CAP.4, ACM_SCP.2)
11. Life-Cycle Documentation (ALC_DVS.1, ALC_LCD.1, ALC.TAT.1)
12. Delivery and Operation Documents (ADO_DEL.2, ADO_IGS.1)
13. Guidance Documents (AGD_ADM.1)
14. Covert channel analysis (AVA_CCA.1)
15. Strength of TOE Security Functions (AVA_SOF.1)
16. Misuse Documentation (AVA_MSU.3)
17. Vulnerabilities Documentation (AVA_VLA.4)
18. Flaw Remediation Documentation (ALC_FLR.2)

7. PP CLAIMS

7.1. Statement of PP Compliance

This Security Target complies with the CWA 14167-2 version 0.28 dated 27 October 2003 Cryptographic Module for CSP Signing Operations with Backup (CMCSOB) Protection Profile (PP).

7.2. Identification of IT Security Requirements Satisfying the PP

The IT requirements that satisfy the PP operations are those contained within section 5.1 TOE Security Functional Requirements. The requirements listed below have either been completed as required by the PP or refined as indicated.

FAU_GEN.1 Audit data generation – refined as follows:

FAU_GEN.1.1

The auditable event Start-up and shutdown of the audit functions is not applicable as these are always present.

Audit data generated by the TOE is in raw form and requires interpretation by an application in the TOE environment before it can be included in a security audit trail.

Unless invoked by an authorised user, results of abstract machine tests and self-tests are only reported in the case of a failure.

Events "Shutdown of the TOE", "Management of TSF data (FMT_MTD.1/Access Control", "Management of TSF data (FMT_MTD.1/Audit" and "Failure with preservation of secure state (FPT_FLS.1" have been deleted for the reasons stated in the footnotes.

FAU_GEN.1.2

Date and time of the event shall be given by the sequence data correlated to time of export of the audit data to the TOE environment.

The audit data for the Crypto Officer and Crypto User roles may only be sufficient to identify the client application. Further refinement of audit data might be provided by audit functions in the TOE environment distinguishing between end-users using the services of the client application.

FAU_STG.2 (TOE) Guarantees of audit data availability – Assignment of "metric for saving audit records" completed as shown below:

FAU_STG.2.3 (TOE) The TSF shall ensure that **60 kB of** audit records will be maintained when the following conditions occur: audit storage exhaustion.

FCS_CKM.1 Cryptographic key generation – Assignment of "cryptographic key generation algorithms", "cryptographic key sizes" and "list of standards" completed as shown below:

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with *the* specified cryptographic key generation *algorithms* **listed below** and specified cryptographic key sizes **specified for each algorithm** that meet the following **standards noted for each algorithm**:

(1) RSA 1024, 2048, 4096 bits key pairs in accordance with ANSI X9.31.
(2) TDES 112, 168 bits in accordance with FIPS PUB 46-3 and ANSI X9.52.
(3) AES 128, 192, 256 bits in accordance with FIPS PUB 197.
(4) DSA 1024 bits key pairs in accordance with FIPS PUB 186-2.
(5) ECDSA in accordance with FIPS PUB 186-2 and ANSI X9.62.

FCS_CKM.2 (BACKUP) Cryptographic key distribution – Assignment of "list of standards" completed as shown below:

FCS_CKM.2.1 (BACKUP) The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **key entry** that meets the following: **TDES 168 bits**.

FCS_CKM.4 Cryptographic key destruction – Assignment of "cryptographic key destruction method" and "list of standards" completed as shown below:

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method **logical or physical (by overwriting) deletion of the memory space** that meets the following: **FIPS 140-2 Level 3.**

FCS_COP.1 (SIGN) Cryptographic operation - Digital signature – Assignment of "cryptographic algorithms", "cryptographic key sizes" and "list of standards" completed as shown below:

FCS_COP.1.1 (SIGN) The TSF shall perform **digital signature generation and verification** in accordance with *the* specified cryptographic *algorithm*s **listed below** and cryptographic key sizes **specified for each algorithm** that meet the following: **standards noted for each algorithm**.

(1) RSA 1024 bits, 2048 bits, 4096 bits with SHA-1, SHA-256, SHA-384, SHA-512 (PKCS #1 V1.5, PKCS #1 PSS),
(2) RSA 1024 bits, 2048 bits, 4096 bits with SHA-1 (FIPS PUB 186-2/ANSI X9.31),
(3) Signature generation/verification DSA 1024 bits with SHA-1 (FIPS PUB 186-2),
(4) Signature generation/verification ECDSA with SHA-1 (FIPS PUB 186-2 Appendix 6 recommended curves).

FCS_COP.1 (BACKUP_ENC) Cryptographic operation – Assignment of "cryptographic algorithms", "cryptographic key sizes" and "list of standards" completed as shown below:

FCS_COP.1.1 (BACKUP_ENC) The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm **TDES** and cryptographic key sizes **168 bits** that meet the following: **FIPS PUB 46-3**.

The portions of the refinement added by the PP that deal specifically with manual key entry have been deleted because they do not apply to the TOE.

FCS_COP.1 (BACKUP_INT) Cryptographic operation – Assignment of "cryptographic algorithms", "cryptographic key sizes" and "list of standards" completed as shown below:

FCS_COP.1.1 (BACKUP_INT) The TSF shall perform **calculation and verification of cryptographic checksums** in accordance with a specified cryptographic algorithm **SHA-1** and cryptographic key sizes **N/A** that meet the following: **FIPS PUB 180-2**.

The refinement added by the PP has been deleted and an explanatory Application Note added, as follows:

Refined by adding:

~~The cryptographic checksum for backup data shall use a backup key and shall be based on symmetric cryptographic algorithms (e.g. keyed hash) or asymmetric cryptographic algorithms (e.g. digital signatures).~~

Application Note:

The SHA-1 checksum is calculated by the TSF and by the equivalent function within the backup token. It can be compared before and after backup and before and after recovery to validate integrity. Because it is calculated within the TSF and backup token at the time it is requested, there is no chance of substituting an illegitimate key value and corresponding digest value. A keyed hash or digital signature is, therefore, not required.

FCS_RND.1 Quality metrics for random numbers – Assignment of the quality metric is shown below:

FCS_RND.1.1 The TOE shall provide a mechanism for generating random numbers that meet **the requirements of ETSI SR 002 176 V1.1.1, dated 2003-03.**

Two elements have been added to the assignment made by the PP for the purpose of the random numbers, as follows:

> **FCS_RND.1.2** The TOE shall be able to enforce the use of TOE-generated random numbers for **FCS_CKM.1, authentication data generation and random number generation.**

FDP_ACF.1 (CRYPTO) Security attribute based access control – The Application Note has been re-worded as shown below to reflect the actual TOE behaviour.

> PP wording: The dual person control requires two users to be authenticated with different identities and with the same role Crypto Officer at the same time.
>
> ST wording: The dual person control requires two users to be authenticated – one with the role Crypto Officer and one with the role Security Officer.

FDP_ACF.1 (BACKUP) Security attribute based access control – One part of the assignment made by the PP, "to enter backup keys (FCS_CKM.2)", has been deleted and an explanatory Application Note added as follows:

> The CKM.2 key entry requirement is met in the TOE by key negotiation between the TSF and the backup token. Therefore, there is no user involvement in key entry.

The following assignment has been made:

> **FDP_ACF.1.3 (BACKUP)** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **User with security attribute Role Crypto Officer is allowed under dual person control to backup all partition objects.**

FDP_BKP.1 Backup and recovery – The following element has been deleted from the PP text because it does not apply to the TOE:

> the cryptographic key(s) needed to verify the cryptographic checksum of the backup data.

The following Application Note has been added:

> The SHA-1 checksum is calculated by the TSF and by the equivalent function within the backup token. It can be compared before and after backup and before and after recovery to validate integrity. Because it is done within the TSF and backup token, there is no chance of substituting an illegitimate key value and corresponding digest value. A keyed hash or digital signature is, therefore, not required.

FDP_ETC.1 Export of user data without security attributes – "Token Access Control (TAC) SFP" has been added as an assignment. The SFR has been refined by adding:

> *All user data whose CKA_SENSITIVE attribute is set is exported in encrypted form via the wrap command.*

FDP_IFF.4 (BACKUP) Partial elimination of illicit information flows – Assignments have been completed as follows:

- [assignment: *other relevant side-channels*] is completed as [assignment: **none**}
- [assignment: *maximum capacity*] is completed as **maximum 0 bits/sec for covert channels and resistance to an attacker with attack potential 'high' for side channels**.

FDP_IFF.4 (CRYPTO) Partial elimination of illicit information flows – Assignments have been completed as follows:

- [assignment: *other relevant side-channels*] is completed as [assignment: **none**}
- [assignment: *maximum capacity*] is completed as **maximum 0 bits/sec for covert channels and resistance to an attacker with attack potential 'high' for side channels**.

FDP_RIP.1 Subset residual information protection – The assignment has been added to as shown below (additional elements are italicised):

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the de-allocation of the resource from the following objects: ***RAD, private keys, in particular* CSP-SCD, *and secret keys***.

FIA_AFL.1 Authentication failure handling – This SFR has two instantiations in the ST, FIA_AFL.1 (SO) and FIA_AFL.1 (User). Assignments have been made in each instantiation as follows:

FIA_AFL.1.1 (SO) The TSF shall detect when **three (3)** unsuccessful authentication attempts occur related to **Security Officer authentication**.

FIA_AFL.1.1 (User) The TSF shall detect when **an *SO* configurable positive integer within the range of three (3) to ten (10)** unsuccessful authentication attempts occur related to **User authentication**.

FIA_ATD.1 User attribute definition – The assignment has been added to as shown below (additional attributes are italicised):

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users:
User ID number
User checkword (RAD)
User role
User failed login count
User "locked" flag.

FIA_SOS.1 Verification of secrets – Assignment has been completed as follows:

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet **the minimum length established by the TOE for each authentication secret**.

FIA_UAU.1 Timing of authentication – The wording of the assignment has been changed for readability and some elements have been added to the assignment as shown below (additional elements are italicised):

FIA_UAU.1.1 The TSF shall allow **the following actions** on behalf of the user to be performed before the user is authenticated:

- **Perform start-up, self-test (FPT_TST.1), detection of the secure blocking state (FPT_FLS.1), detection of violation of physical integrity (FPT_PHP.2),**
- ***Perform basic diagnostic functions, such as checking the communications from the host to the card, checking firmware level and token info and checking information on mechanisms supported.***
- ***Open a session***
- ***Access Public data objects***
- **Identification (FIA_UID.1).**

FIA_UID.1 Timing of identification – The wording of the assignment has been changed for readability and some elements have been added to the assignment as shown below (additional elements are italicised):

FIA_UID.1.1 The TSF shall allow **the following actions** on behalf of the user to be performed before the user is identified:

- **Perform start-up, self-test (FPT_TST.1), detection of the secure blocking state (FPT_FLS.1), detection of violation of physical integrity (FPT_PHP.2),**
- **Perform basic diagnostic functions, such as checking the communications from the host to the card, checking firmware level and token info and checking information on mechanisms supported.**

FMT_MSA.1 (ROLE_CRYPTO) Management of security attributes – The assignment was completed by specifying [assignment: other operations] as [assignment: **none**].

FMT_MSA.1 (ROLE_AUDIT) Management of security attributes – The assignment was completed by specifying [assignment: other operations] as [assignment: **none**].

FMT_MTD.1 (Access Control) Management of TSF data – This SFR does not apply in the case of the TOE. An explanatory Application Note was added.

FMT_MTD.1 (USER_Crypto) Management of TSF data – This SFR was refined as shown below to change the role that is capable of performing the specified operations from Crypto Officer to Security Officer, a more restrictive role.

> **FMT_MTD.1.1 (USER_CRYPTO)** The TSF shall restrict the ability to change default *and* delete the **Identity and RAD for user with role attribute Crypto Officer and Crypto User** to ***Security Officer***.

FMT_MTD.1 (RAD) Management of TSF data – This SFR was refined as shown below to change the role that is capable of performing the specified operations from "User for its own RAD" to "Crypto Officer for the identity associated with the RAD", a more restrictive role.

> **FMT_MTD.1.1 (RAD)** The TSF shall restrict the ability to modify the **RAD** to ***Crypto Officer for the identity associated with the* RAD**.

FMT_MTD.1 (AUDIT) Management of TSF data – The Crypto Officer role has been added to the roles capable of performing the specified operations because restricting the ability to query audit data to only the Auditor role contradicts **FDP_ACF.1.2 (AUDIT).**

FMT_SMR.1 Security roles – The Security Officer role was added to the list of roles maintained by the TOE.

FPT_AMT.1 Abstract machine testing – The condition "during initial startup" was added to the assignment made by the PP.

FPT_ITI.1 Inter-TSF detection of modification – The assignment made by the PP in FPT_ITI.1.1 has been changed from "cryptographic checksum according to the list of approved algorithms and parameters" to SHA-1 digest to make the algorithm choice explicit. The wording of the assignment made by the PP in FPT_ITI.1.2 was changed from "alarm indication" to "error indication" to more accurately reflect the TOE behaviour. The following text has been added to the Application Note for this SFR:

> The SHA-1 checksum is calculated by the TSF and by the equivalent function within the backup token. It can be compared before and after backup and before and after recovery to validate integrity. Because it is done within the TSF and backup token, there is no chance of substituting an illegitimate key value and corresponding digest value. A keyed hash or digital signature is, therefore, not required.
>
> FPT_ITC.1 and FPT_ITI.1 are included because they are specified by the PP. However, the requirement for confidentiality and integrity protection of the backup data is best stated by FTP_ITC.1, which is included as an addition to the PP.

FPT_PHP.2 Notification of physical attack – The following text was added to the Refinement statement for this SFR:

> *Note that the TOE does not have a door or removable cover. Opening the TOE would, therefore, require the use of tools that would necessitate the removal of the TOE from its intended operating environment.*

FPT_RCV.1 Manual recovery – The assignment was completed as shown below:

> **FPT_RCV.1.1** After **a self-test failure**, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT_TST.1 TSF testing – The selection "at the conditions installation and maintenance" has been removed because there are no states of the TOE that correspond to installation and maintenance.

FTP_TRP.1 Trusted path – The SFR statement in FTP_TRP.1.1 has been refined to "*physically and* logically distinct" to reflect the actual nature of the TOE's trusted path device. The following additional functions have been added in FTP_TRP.1.3:

- Upload of the TSF-generated authentication data to the iKey
- Upload of the TSF-generated challenge secret to the PED display
- Entry of M of N Activation secret shares
- Entry of the Token Cloning Domain key

7.3. Identification of Security Objectives and IT Security Requirements Additional to the PP

7.3.1. Security Objectives

The following objectives have been added or represent refinements of the PP objectives.

O.Admin – has been added to address the need for secure management of the TOE.

O.Approved_Algorithms – is a refinement of the PP objectives in that the statement requiring use of approved algorithms is found in three of the PP objectives and it has simply been pulled out to stand as its own objective statement.

O.Auth_Data_Protect – has been added to counter T.PIN_Compromise.

O.Backup and O.Data_Exchange_Protect – refine O.Protect_Exported_Data by setting separate objectives for backup and for the export of user symmetric keys and data for data exchange purposes.

O.Import_Code – has been added to counter T.Bad_FW_Load.

O.Key_Secure – expands the objective O.CSP-SCD_Secure to include all key material handled by the TOE.

O.Multi-Person_Control – has been added to assist in countering T.Key_Management and T.Management.

O.Secure_Init – has been added to assist in countering many of the threats by ensuring that the TOE is always in a secure state when it starts up.

O.Self_Protect – has been added to assist in countering many of the threats by ensuring that the TSF are protected from external influence and cannot be bypassed.

O.User_Data_Protect – has been added to address the protection of user data other than key material.

7.3.2. IT Security Requirements

The IT Security Requirements for the TOE additional to those of the PP are all collected in section 5.2.

The IT Security Requirements for the Environment additional to those of the PP are all collected in section 5.3.5.

8. RATIONALE

The mappings and descriptions of the rationale for the Security Objectives, IT Security Requirements and Dependencies, and Assurance Measures are presented in the tables at the end of this section.

8.1. Security Objectives Rationale

The tables Table 8-1, Table 8-2 and Table 8-3 demonstrate the necessity of the security objectives and their appropriateness in countering the stated threats and providing for the stated assumptions.

8.2. IT Security Requirements Rationale

Table 8-4 shows the necessity of the Security Functional Requirements and Table 8-5 maps Security Functional Requirements to Security Objectives and provides the rationale that the SFRs, singly or in combination, meet the Security Objectives. Table 8-6 demonstrates that all dependencies for the SFRs have been met. Table 8-7 maps IT Security Functions to IT Security Requirements and Security Functional Requirements. This shows the necessity of each of the IT Security Functions presented in the TOE Summary Specification. Table 8-8 maps Security Functional Requirements to IT Security Functions and presents the rationale for why each IT Security Function satisfies the requirements of a Security Functional Requirement.

8.2.1. Explicitly Stated Security Requirements

This Security Target specifies two explicitly stated Security Functional Requirements. The two explicitly stated requirements with the rationale for their inclusion are as follows:

FCS_RND.1 – This requirement is drawn from the document “Functionality classes and evaluation methodology for true random number generators” (AIS 31) [8]. It is used in this ST because of the importance of high quality random numbers to the process of key and key pair generation and for TSF-generated secrets.

FDP_BKP.1 – This requirement is explicitly stated in this ST to address a unique requirement for the TOE to be capable of performing a secure backup of cryptographic material that can be used in the recovery of the host processing environment.

8.2.2. Appropriateness of Strength of Function

The minimum Strength of Function requirement is SoF-high. This is above the SoF-basic required for the EAL 4 assurance requirements that assume a low attack potential. It is considered to be appropriate for the TOE security functions given its assumed (and intended) operating environment because attackers can be assumed to have low to moderate/high attack potential, as described in section 3.3. The explicit Strength of Function claims of SoF-high for the authentication mechanisms and the Pseudo-Random Number Generator are considered to be necessary given the intended role of the TOE as the hardware cryptographic module in what is typically a critical infrastructure system.

8.2.3. Appropriateness of Assurance Requirements

The assurance requirements chosen for the TOE, EAL 4 augmented by ADV_IMP.2, ALC_FLR.2, AVA_CCA.1, AVA_MSU.3 and AVA_VLA.4 are considered to be appropriate for the TOE in its assumed (and intended) operating environment for the following reasons:

1. There are specific customer requirements for Certification Authority (CA) or Certification Service Provider (CSP) components that meet the EAL 4 assurance requirements. The TOE, as part of a larger CA or CSP system, must meet the EAL 4 requirements at a minimum, but does not need to exceed them.

2. Because the CA and CSP systems, for example, are critical infrastructure systems, customers require a relatively high level of assurance that the components that make them up have been developed and are maintained using sound engineering security practices.
3. It is assumed that, for most of its life-cycle, the TOE will be contained within a larger secure environment. It will, therefore, not be exposed to a threat environment that allows easy access by highly capable outsiders. The main exception to this is when it is in transit when it will be in a state that is either zeroized or where all of its sensitive data will be encrypted using Triple DES encryption. Thus, the assumption of moderate attack potential for outsiders is considered appropriate.
4. Although the TOE will normally be contained within a secure environment, the potential value of the key material stored within the TOE may be sufficient to result in insider attacks. Because insiders would typically have access to the TOE or components of it and would be likely to have detailed knowledge of the TOE and its configuration in their environment, they are considered to have High attack potential.
5. The augmentation of including ALC_FLR.2 is in response to existing company practice that has been implemented to meet customer requirements for flaw reporting and fixing.
6. The augmentation of including AVA_CCA.1, AVA_MSU.3 and AVA_VLA.4 is in response to the attack potentials described in sub-paragraphs 3 and 4 above. The augmentation of including ADV_IMP.2 is made because it is a dependency of AVA_CCA.1.

8.2.4. Applicability and Appropriateness of Assurance Requirements for Explicitly Stated Requirements

The explicitly stated requirements, FCS_RND.1, specifies a metric to be achieved with Strength of Function High for the RNG employed by the TOE. AVA_SOF.1 and AVA_VLA.4, as an augmentation to the EAL 4 package are directly applicable in demonstrating that the TOE meets the required metric. These are supported by the remaining Assurance Requirements within the EAL 4 package and the specified augmentations. FDP_BKP.1 requires that key material be protected during the backup and recovery process. The rationale presented in section 8.2.3 above applies to this SFR.

8.3. Assurance Measures

Table 8-9 – Assurance Measures shows each of the security assurance requirements of the TOE and maps each to the applicable assurance evidence provided for the evaluation.

Table 8-1 – Necessity of Security Objectives

Objective	Necessitated by:
O.Admin	T.Management
O.Approved_Algorithms	P.Algorithms
O.Audit_CM	T.Insecure_Init, T.Key_Management, T.Management, T.Misuse_Sign
O.Auth_Data_Protect	T.PIN_Compromise
O.Backup	T.CSP_SCD_Disclose, T.CSP_SCD_Distortion, T.Management
O.Check_Operation	T.Bad_FW_Load, T.CSP_SCD_Disclose, T.CSP_SCD_Distortion, T.Malfunction, T.Phys_Manipul, T.Unauth_Function
O.Control_Access	T.Bad_FW_Load, T.Insecure_Init, T.Key_Management, T.Management, T.Misuse_Sign
O.Data_Exchange_Protect	T.Exchange
O.Detect_Attack	T.CSP_SCD_Distortion, T.Phys_Manipul
O.Error_Secure	T.CSP_SCD_Distortion, T.Malfunction, T.Phys_Manipul, T.Unauth_Function
O.Import_Code	T.Bad_FW_Load
O.Key_Secure	T.CSP_SCD_Derive, T.CSP_SCD_Disclose, T.CSP_SCD_Distortion, T.Key_Management, T.Phys_Manipul
O.Multi-Person_Control	T.Key_Management, T.Insecure_Init, T.Management, T.Misuse_Sign
O.Secure_Init	T.Insecure_Init, T.Malfunction
O.Self_Protect	T.Malfunction, T.Unauth_Function
O.Sign_Secure	T.CSP_SCD_Derive, T.CSP_SCD_Disclose, T.Signature_Forgery
O.User_Authentication	T.Key_Management, T.Management, T.Misuse_Sign
O.User_Data_Protect	T.CSP_SCD_Disclose, T.CSP_SCD_Distortion, T.Misuse_Sign
O.ENV_Application	A.Correct_DTBS, A.User_Authentication, T.Data_Manipul
O.ENV_Audit	A.Audit_Support
O.ENV_AuthData	A.Admin, A.User_Authentication, A.User_Management
O.ENV_Backup	A.Data_Store
O.ENV_Human_Interface	A.Human_Interface, A.User_Authentication
O.ENV_Outage_Protection	T.Malfunction
O.ENV_Personnel	T.Insecure_Oper, A.Admin, A.Audit_Support, A.User_Management
O.ENV_Protect_Access	T.Insecure_Oper, A.Controlled_Access
O.ENV_Recovery	T.Insecure_Init, T.Malfunction, A.Data_Store
O.ENV_Secure_Init	T.Insecure_Init, A.Data_Store
O.ENV_Secure_Oper	T.Insecure_Oper, A.Data_Store, A.Correct_DTBS, T.Data_Manipul
O.ENV_Signed_FW_Update	A.Legitimate_FW_Update

Table 8-2 – Mapping of Objectives to Threats

Threats	Objectives	Rationale
T.Bad_FW_Load	O.Check_Operation, O.Control_Access, O.Import_Code	This combination of objectives counters the threat by ensuring that the TOE will control the loading of firmware code, will load only valid firmware images and will check to verify that the integrity of the code is preserved prior to each activation of the code.
T.CSP_SCD_Derive	O.Key_Secure, O.Sign_Secure	O.Key_Secure is responsible to ensure that no information about keys, such as the CSP-SCD, is directly transmitted to any entity outside the TOE. O.Sign_Secure ensures that the algorithms and the specific implementation will not reveal the CSP-SCD.
T.CSP_SCD_Disclose	O.Backup, O.Check_Operation, O.Key_Secure, O.Sign_Secure, O.User_Data_Protect	Unencrypted export of the CSP_SCD is prohibited by O.Key_Secure and O.Backup, and the incorrect operation is addressed by O.Check_Operation. In addition O.Sign_Secure ensures that the CSP-SCD is not disclosed as part of the signed data exported to the user. O.User_Data_Protect assists by ensuring that user data in all forms, including residual data, is protected from disclosure.
T.CSP_SCD_Distortion	O.Backup, O.Check_Operation, O.Detect_Attack, O.Error_Secure, O.Key_Secure, O.User_Data_Protect	O.Check_Operation will ensure that the TOE will check the CSP-SCD regularly. O.Error_Secure will prevent the TOE to use distorted CSP-SCD after it has detected the distortion and O.Detect_Attack will prohibit the use of a distorted CSP-SCD after a physical attack (of course in the case of a physical attack the TOE will itself destroy the CSP-SCD and enter a state where it can only be reused after a secure re-initialisation). O.Backup addresses the integrity and confidentiality protection measures to CSP-SCD when they are exported from the TOE e.g. for the purpose of backup and restore. O.User_Data_Protect assists by allowing the user to verify the authenticity of key material before it is used.
T.Exchange	O.Data_Exchange_Protect	This objective counters the threat by ensuring that the TOE has the capability to protect data exchanges from unauthorised disclosure and modification.

Threats	Objectives	Rationale
T.Insecure_Init	O.Audit_CM, O.Control_Access, O.Multi-Person_Control, O.Secure_Init, O.ENV_Recovery, O.ENV_Secure_Init	This threat is countered by O.Key_Secure with respect to the secure CSP-SCD generation and management, O.Control_Access with respect to the unauthorised use of services (also in the initialization phase) as well as by objectives on the TOE environment O.ENV_Secure_Init and O.ENV_Recovery. O.Secure_Init ensures that the TOE assumes its initial secure state immediately upon power-up, reset, or after other restart conditions. In addition, O.Multi-Person_Control provides the ability to enforce multi-person control to ensure that everything is correct before initialization is performed and O.Audit_CM provides the ability to check if the initialization process has been performed correctly.
T.Key_Management	O.Audit_CM, O.Control_Access, O.Key_Secure, O. Multi-Person_Control, O.User_Authentication	This combination of objective counters the threat by ensuring that keys can only be generated and managed through their life-cycle to destruction on the module using secure techniques (O.Key_Secure). O.Control_Access supports this by ensuring that access to key management functions is only granted to authorised users. O User_Authentication and O.Multi-Person_Control support this. O.Audit_CM provides the ability to check that key management functions are used properly.
T.Malfunction	O.Check_Operation, O.Error_Secure, O.Secure_Init, O.Self_Protect, O.ENV_Outage_Protection, O.ENV_Recovery	This threat is countered by O.Check_Operation and O.Error_Secure (which ensures that the TOE will not continue to operate with the CSP-SCD when it has detected a malfunction). Due to the criticality of the TOE and the requirement for resistance to physical attacks, maintenance of the TOE is also critical and repairing the TOE might be impossible without deleting the CSP-SCD. The combination of O.Secure_Init and O.Self_Protect ensures that the TOE will start in a secure state and will protect itself from deliberate attempts to subvert its security enforcement by inducing faults. The TOE should also be protected as far as possible from defects caused by accidental mishandling amd environmental failures (this is covered by the objective O.ENV_Outage_Protection). On the other hand, if a defect occurs, procedures within the TOE environment have to exist that allow the organisation operating the TOE to recover in a secure way from this defect. This is covered by the objective O.ENV_Recovery.

Threats	Objectives	Rationale
T.Management	O.Admin, O.Audit_CM, O.Backup, O.Control_Access, O.Multi-Person_Control, O.User_Authentication	This threat is countered by O.Control_Access, which restricts the use of TOE management functions to authorised users, O.User_Authentication and O.Multi-Person_Control, which ensures that invoking a management function has the authorisation and O.Audit_CM, which allows to trace the actions of those users. In addition the objective O.Backup prohibits the modification of data exported by the TOE when it is imported again (which otherwise could be used to manipulate TSF management data).
T.Misuse_Sign	O.Audit_CM, O.Control_Access, O.Multi-Person_Control, O.User_Authentication, O.User_Data_Protect	O.Control_Access counters this threat for the user known to the TOE. O.User_Authentication and O.Multi-Person_Control prevents the misuse by persons not authorised to use the TOE and O.Audit_CM allows checking, if an unauthorised user has attempted to get access to the TOE or if an authorized user has attempted to misuse the TOE by attempting to use functions he is not allowed to use. O.User_Data_Protect assists by allowing the user to verify the authenticity and the correctness of key material before it is used.
T.Phys_Manipul	O.Detect_Attack, O.Check_Operation, O.Error_Secure, O.Key_Secure, O.ENV_Protect_Access	O.Detect_Attack counters this threat as long as the TOE is directly able to detect that it is under attack. This includes manipulation by authorised users. O.Check_Operation counters the case where the TOE does not detect the physical manipulation directly but detects an error during operation that might have been caused by a physical attack. O.Error_Secure and O.Key_Secure enforce a secure state of the TOE if such error is detected, particularly for key protection. Since it is obvious that the TOE is not able to withstand all kind of physical manipulation, O.ENV_Protect_Access shall prohibit (as far as possible) the likelihood that an attacker is able to perform any physical manipulation on the TOE.
T.PIN_Compromise	O.Auth_Data_Protect	This objective counters the threat by ensuring that users' authentication data used for authentication cannot be easily guessed or captured via the host computer. Any attempt to impersonate a legitimate user will also be made significantly more difficult by imposing a limit on the number of authentication attempts before the user is locked or erased.
T.Signature_Forgery	O.Sign_Secure	This objective counters the threat by ensuring that the TOE employs signature creation mechanisms that are technically impractical to compromise and that protect the security of the private key.

Threats	Objectives	Rationale
T.Unauth_Function	O.Check_Operation, O.Error_Secure, O.Self_Protect	This combination of objectives counters the threat by ensuring that the TOE provides self-protection capability plus ensuring that error conditions do not leave the TOE is a non-secure state. This ensures that unauthorised functions cannot be executed either by subverting the TOE or by introducing malicious code through exploitation of errors such as buffer overflows.
T.Data_Manipul	O.ENV_Application, O.ENV_Secure_Oper	This objective counters the threat by ensuring that the application in the host environment can be trusted to operate correctly and not manipulate data, such as the DTBS, that is passed to the TOE.
T.Insecure_Oper	O.ENV_Personnel, O.ENV_Protect_Access, O.ENV_Secure_Oper	This threat is addressed by the objective O.ENV_Secure_Oper. Physical protection of the TOE, which is also necessary to operate the TOE securely, is addressed by O.ENV_Protect_Access. In addition all personnel performing operational activities with the TOE or within the TOE environment must be aware of their duties and responsibilities and must be trained to perform their actions in accordance with the defined procedures. This is addressed by the objective O.ENV_Personnel.

Table 8-3 – Mapping of Objectives to Assumptions and Policies

Assumptions	Objectives	Rationale
A.Admin	O.ENV_AuthData, O.ENV_Personnel	O.ENV_Personnel satisfies the assumption by providing for competent administrators for the TOE and O.ENV_AuthData ensures that authentication data for the administrators is properly protected, thus ensuring that only the proper administrators have access to the administrative functions.
A.Audit_Support	O.ENV_Audit, O.ENV_Personnel	This combination of objectives satisfies the assumption by ensuring that the environment provides adequate audit processing and review to support the secure operation of the TOE and that there are competent personnel to review and manage the audit data.
A.Controlled_Access	O.ENV_Protect_Access	This objective satisfies the assumption by ensuring that adequate physical security and procedural measures are in place to control access to the facility hosting the TOE. This protects against attacks requiring direct physical access to the TOE or host system and leakage of information via monitoring the power consumption or via radiation.

Assumptions	Objectives	Rationale
A.Correct_DTBS	O.ENV_Application, O.ENV_Secure_Oper	O_ENV_Application ensures that the applications that use the TOE will perform the required checks on the data they pass to the TOE. O.ENV_Secure_Oper ensures that the necessary operational procedures are in place for the organisation operating the TOE as part of their certification system. With the sum of these objectives the assumption is covered.
A.Data_Store	O.ENV_Backup, O.ENV_Recovery, O.ENV_Secure_Init, O.ENV_Secure_Oper	This combination of objectives satisfies the assumption by ensuring that the TOE environment protects the integrity and availability of data required for TOE initialisation, start-up, operation and recovery if stored or handled outside the TOE.
A.Human_Interface	O.ENV_Application, O.ENV_Human_Interface	The client application will provide the human interface for interaction with the TOE and ensures the confidentiality and integrity of data passed to the TOE..
A.Legitimate_FW_Update	O.ENV_Signed_FW_Update	This objective satisfies the assumption by ensuring that a process is in place within the environment to digitally sign firmware update packages to prove their legitimacy.
A.User_Authentication	O.ENV_Application, O.ENV_AuthData, O.ENV_Human_Interface	This combination of objectives satisfies the assumption by ensuring that the environment provides an application that interacts correctly with human users for the purpose of authentication, protects the authentication data provided by the users and represents those users correctly to the TOE. Also, that the human users exercise proper control over their authentication data.
A.User_Management	O.ENV_AuthData, O.ENV_Personnel	This combination of objectives satisfies the assumption by ensuring that the management policies and procedures governing the assignment of individual human users to roles on the TOE are properly followed and that the users protect their authentication data so as to support their role assignments.
P.Algorithms	O.Approved_Algorithms	This objective satisfies the assumption by ensuring that the TOE provides the algorithms required by organizational policy.

Table 8-4 – Necessity of Security Functional Requirements

SFR	Necessitated by:
FAU_GEN.1	O.Audit_CM, O.Backup, O.Check_Operation
FAU_GEN.2	O.Audit_CM, O.Backup
FAU_STG.2	O.Audit_CM
FCS_CKM.1	O.Approved_Algorithms, O.Key_Secure
FCS_CKM.2 (BACKUP)	O.Approved_Algorithms, O.Key_Secure, O.Backup
FCS_CKM.4	O.Approved_Algorithms, O.Key_Secure
FCS_COP.1 (SIGN)	O.Approved_Algorithms, O.Key_Secure, O.Sign_Secure
FCS_COP.1 (BACKUP_ENC)	O.Approved_Algorithms, O.Backup
FCS_COP.1 (BACKUP_INT)	O.Approved_Algorithms, O.Backup
FCS_RND.1	O.Approved_Algorithms, O.Key_Secure
FDP_ACC.1 (CRYPTO)	O.Control_Access, O.Key_Secure
FDP_ACC.1 (AUDIT)	O.Audit_CM, O.Control_Access
FDP_ACC.1 (BACKUP)	O.Backup, O.Control_Access
FDP_ACF.1 (CRYPTO)	O.Control_Access, O.Key_Secure
FDP_ACF.1 (AUDIT)	O.Audit_CM, O.Control_Access
FDP_ACF.1 (BACKUP)	O.Backup, O.Control_Access
FDP_BKP.1	O.Backup, O.Key_Secure
FDP_ETC.1	O.Data_Exchange_Protect
FDP_IFC.1 (BACKUP)	O.Backup. O.Key_Secure, O.Sign_Secure
FDP_IFC.1 (CRYPTO)	O.Key_Secure, O.Sign_Secure
FDP_IFF.4 (BACKUP)	O.Backup. O.Key_Secure, O.Sign_Secure
FDP_IFF.4 (CRYPTO)	O.Key_Secure, O.Sign_Secure
FDP_RIP.1	O.Key_Secure, O.User_Data_Protect
FDP SDI.2	O.Key_Secure, O.User_Data_Protect
FIA_ATD.1	O.User_Authentication
FIA_UID.1	O.User_Authentication
FIA_UAU.1	O.User_Authentication
FIA_AFL.1 (SO)	O.User_Authentication, O.Auth_Data_Protect
FIA_AFL.1 (User)	O.User_Authentication, O.Auth_Data_Protect
FIA_SOS.1	O.User_Authentication, O.Auth_Data_Protect
FMT_MSA.1 (ROLE_CRYPTO)	O.Backup, O.Control_Access
FMT_MSA.1 (ROLE_AUDIT)	O.Control_Access
FMT_MSA.2	O.Control_Access
FMT_MSA.3	O.Backup, O.Control_Access
FMT_MTD.1 (Access Control)	O.Control_Access
FMT_MTD.1 (AUDIT)	O.Audit_CM
FMT_MTD.1 (RAD)	O.User_Authentication
FMT_MTD.1 (USER_Crypto)	O.User_Authentication
FMT_MTD.1 (USER_Audit)	O.User_Authentication
FMT_SMF.1	O.Admin, O.Audit_CM, O.Control_Access, O.User_Authentication
FMT_SMR.1	O.Control_Access
FPT_AMT.1	O.Check_Operation, O.Error_Secure, O.Secure_Init
FPT_FLS.1	O.Error_Secure, O.Secure_Init
FPT_ITC.1	O.Backup
FPT_ITI.1	O.Audit_CM, O.Backup
FPT_PHP.2	O.Detect_Attack
FPT_PHP.3	O. Detect_Attack

SFR	Necessitated by:
FPT_RCV.1	O.Error_Secure
FPT_TST.1	O.Check_Operation, O.Control_Access, O.Error_Secure , O.Secure_Init
FTP_TRP.1	O.Auth_Data_Protect, O.User_Authentication
Additions to the PP	
FCS_CKM.2 (FW Update)	O.Import_Code
FCS_CKM.3	O.Key_Secure
FCS_COP.1 (DIGEST)	O.Approved_Algorithms
FCS_COP.1 (RSA ENC/DEC)	O.Approved_Algorithms, O.Data_Exchange_Protect
FCS_COP.1 (TDES ENC/DEC)	O.Approved_Algorithms, O.Data_Exchange_Protect, O.Import_Code
FCS_COP.1 (AES ENC/DEC)	O.Approved_Algorithms, O.Data_Exchange_Protect
FDP_ACC.1 (TAC)	O.Control_Access, O.Data_Exchange_Protect, O.Key_Secure
FDP_ACF.1 (TAC)	O.Control_Access, O.Data_Exchange_Protect, O.Key_Secure
FDP_DAU.1	O.User_Data_Protect
FDP_DAU.2	O.Sign_Secure, O.User_Data_Protect
FDP_ITC.1	O.Data_Exchange_Protect
FDP_RIP.2	O.User_Data_Protect
FDP_UCT.1	O.Data_Exchange_Protect
FDP_UIT.1	O.Data_Exchange_Protect
FIA_SOS.2	O.User_Authentication, O.Auth_Data_Protect
FIA_UAU.4	O.User_Authentication, O.Auth_Data_Protect
FIA_UAU.5	O.Multi-Person_Control
FIA_USB.1	O.User_Authentication
FMT_MOF.1	O.Admin
FMT_MSA.1 (Object Attributes)	O.Control_Access
FMT_MSA.2 (Object Attributes)	O.Control_Access
FMT_MSA.3 (Object Attributes)	O.Control_Access
FMT_MTD.1 (Login Failures)	O.Admin, O.User_Authentication
FMT_MTD.1 (UAV)	O.Admin, O.User_Authentication
FMT_MTD.1 (SOV)	O.Admin, O.User_Authentication
FMT_SMF.1 (Policies)	O.Admin
FPT_RVM.1	O.Self_Protect
FPT_SEP.1	O.Self_Protect
FRU_FLT.1	O.Error_Secure
FTP_ITC.1 (FW Update)	O.Import_Code
FTP_ITC.1 (Key Cloning)	O.Backup
SFRs for TOE Environment	
FAU_SAR.1 (ENV)	O.ENV_Audit
FAU_STG.1 (ENV)	O.ENV_Audit
FCS_CKM.1 (ENV/FW Update)	O.ENV_Signed_FW_Update
FCS_CKM.2 (ENV/FW Update)	O.ENV_Signed_FW_Update
FCS_CKM.2 (ENV/BACKUP)	O.ENV_Backup
FCS_COP.1 (ENV/ENC FW Update)	O.ENV_Signed_FW_Update
FCS_COP.1 (ENV/SIGN FW Update)	O.ENV_Signed_FW_Update
FCS_COP.1 (ENV/BACKUP_ENC)	O.ENV_Backup
FCS_COP.1 (ENV/BACKUP_INT)	O.ENV_Backup
FDP_ACC.1 (CLIENT)	O.ENV_Application
FDP_ACF.1 (CLIENT)	O.ENV_Application
FDP_ACC.1 (ENV/BACKUP)	O.ENV_Backup
FDP_ACF.1 (ENV/BACKUP)	O.ENV_Backup
FDP_UIT.1	O.ENV_Application

SFR	Necessitated by:
FIA_UAU.1 (CLIENT)	O.ENV_Application, O.ENV_Human_Interface
FIA_UID.1 (CLIENT)	O.ENV_Application, O.ENV_Human_Interface
FTP_ITC.1 (ENV/FW Update)	O.ENV_Signed_FW_Update
FTP_ITC.1 (ENV/Key Cloning)	O.ENV_Backup
FTP_TRP.1 (CLIENT)	O.ENV_Human_Interface

Table 8-5 – Mapping of Security Functional Requirements to Objectives

Objectives	Security Functional Requirements	Rationale
O.Admin	FMT_MOF.1, FMT_MTD.1, FMT_SMF.1, FMT_SMR.1	This combination of SFRs satisfies the objective by requiring that the TOE provide suitable roles and management functions to administer the TOE.
O.Approved_Algorithms	FCS_CKM.1, FCS_CKM.2 (BACKUP), FCS_CKM.3, FCS_CKM.4, FCS_COP.1 (BACKUP_ENC), FCS_COP.1 (BACKUP_INT), FCS_COP.1 (RSA ENC/DEC), FCS_COP.1 (TDES ENC/DEC), FCS_COP.1 (AES ENC/DEC), FCS_COP.1 (SIGN), FCS_COP.1 (DIGEST), FCS_RND.1	This combination of SFRs satisfies the objective by requiring that the TOE provide approved algorithms and key management techniques.
O.Audit_CM	FAU_GEN.1, FAU_GEN.2, FAU_STG.2 (TOE), FDP_ACC.1 (AUDIT), FDP_ACF.1 (AUDIT), FMT_MTD.1 (AUDIT), FMT_SMF.1, FPT_ITI.1	Audit generation is implemented by the SFR FAU_GEN.1 and FAU_GEN.2 with the audit events matching the list in O.Audit_CM. Additional audit is implemented by the SFR FAU_GEN.1 and FAU_GEN.2. The TOE stores the audit data according to the SFR FAU_STG.2 (TOE) until the audit trail is exported upon request of the Auditor or Crypto-officer under control of the SFR FDP_ACC.1 (AUDIT), FDP_ACF.1 (AUDIT) and FMT_MTD.1 (AUDIT). FMT_SMF.1 and FMT_MTD.1 (AUDIT) require management function for the audit. These management functions are provided to the Auditor only. The integrity of the audit data will be ensured by the SFR FAU_STG.2 (TOE) inside the TOE.
O.Auth_Data_Protect	FIA_UAU.4, FIA_AFL.1 (SO), FIA_AFL.1 (User), FIA_SOS.1, FIA_SOS.2, FTP_TRP.1	This combination of SFRs satisfies the objective by requiring that the TOE: • prevent reuse/replay of authentication data (FIA_UAU.4), • prevent brute force attacks on authentication data (FIA_AFL.1) • ensure that authentication data meets the required minimum strength requirements (FIA_SOS.1) • enforce the use of TSF-generated authentication data (FIA_SOS.2), and • utilise a trusted path to protect authentication data from eavesdropping (FTP_TRP.1).

Objectives	Security Functional Requirements	Rationale
O.Backup	FAU_GEN.1, FAU_GEN.2, FCS_CKM.2 (BACKUP), FCS_COP.1 (BACKUP_ENC), FCS_COP.1 (BACKUP_INT), FDP_ACC.1 (BACKUP), FDP_ACF.1 (BACKUP), FDP_ BKP.1, FDP_IFC.1, FDP_IFF.4, FMT_MSA.1 (ROLE_CRYPTO), FMT_MSA.3, FPT_ITC.1, FPT_ITI.1, FTP_ITC.1 (Key Cloning)	The TOE backup and restore functions requires the SFR FDP_BKP.1 the confidentiality and integrity protection of backup data. The backup and restore of CSP-SCD, other user data and TSF data is described in the SFR FDP_BKP.1. The confidentiality and integrity protection of the TSF data as part of the backup data is implemented by the SFR FPT_ITC.1 and SFR FPT_ITI.1 The FDP_BKP.1 needs the cryptographic functions implemented by the following SFR: (i) import the backup keys by FCS_CKM.2, (ii) encryption of backup data by FCS_COP.1/BACKUP_ENC, (iii) data integrity protection by FCS_COP.1 (BACKUP_INT). The SFR FDP_BKP.1 requires encrypting the CSPSCD and electronically exported keys if they are exported. The backup and restore TSF will be under access control required by the SFR FDP_ACF.1 (BACKUP) according to FDP_ACC.1 (BACKUP). The SFR FMT_MSA.1 (ROLE_BACKUP) and FMT_MSA.3 extend the management functions of security attributes to the Backup SFP. The SFR FAU_GEN.1 and FAU_GEN.2 require audit data specific for the use of the backup and restore function associated with the identity of the users. Because FDP_BKP.1 handles and exports the CSP-SCD outside the TSC the TOE shall protect against side-channels to prevent any illicit information flow. The SFR FDP_IFC.1/BACKUP and FDP_IFF.4 (BACKUP) implements this protection. For the TOE, FTP_ITC.1 (Key Cloning) ensures that backup data is protected when transmitted from the TOE to a backup token and vice versa.
O.Check_Operation	FAU_GEN.1, FPT_AMT.1, FPT_TST.1	This security objective is implemented in the TOE by the SFR for abstract machine testing FPT_AMT.1 and TSF testing FPT_TST.1. If these tests detect an error the TOE will transit into a secure state (see O.Error_secure) and prevent the normal operation. FAU_GEN.1 generates audit records about the test results of the SFR FPT-AMT.1 and FPT_TST.1 to inform the user (Auditor or Crypto-officer) about the performed self-tests and their results. The FPT_TST.1 includes checks of the executable code.

Objectives	Security Functional Requirements	Rationale
O.Control_Access	FDP_ACC.1 (CRYPTO), FDP_ACF.1 (CRYPTO), FDP_ACC.1 (BACKUP), FDP_ACF.1 (BACKUP), FDP_ACC.1 (AUDIT), FDP_ACF.1 (AUDIT), FDP_ACC.1 (TAC), FDP_ACF.1 (TAC), FMT_MSA.1 (ROLE_CRYPTO), FMT_MSA.1 (ROLE_AUDIT), FMT_MSA.2, FMT_MSA.3, FMT_MSA.1 (Object Attributes), FMT_MSA.2 (Object Attributes), FMT_MSA.3 (Object Attributes), FMT_MTD.1 (Access Control), FMT_MTD.1 (AUDIT), FMT_SMF.1, FMT_SMR.1	Access control is implemented in the TOE FDP_ACC.1 (CRYPTO), FDP_ACF.1 (CRYPTO), FDP_ACC.1 (BACKUP), FDP_ACF.1 (BACKUP), FDP_ACC.1 (AUDIT), FDP_ACF.1 (AUDIT), FDP_ACC.1 (TAC), FDP_ACF.1 (TAC), with the roles Auditor, Crypto-officer and Crypto-user as defined by the SFR FMT_SMR.1. The SFRs FMT_MSA.1 (ROLE_CRYPTO), FMT_MSA.1 (ROLE_AUDIT), FMT_MSA.2, FMT_MSA.3, FMT_MTD.1 (ACCESS_CONTROL), FMT_MTD.1 (AUDIT) and FMT_SMF.1 assign the management functions for the cryptographic to the Crypto-officer and audit functions to the Auditor. The SFR FMT_MSA.1 (ROLE_CRYPTO) extend the Crypto-officer's management functions to backup and restore. The SFR require the TSF to enforce the Audit-SFP, Backup- SFP and Crypto-SFP to provide restrictive default values for security attributes which may be changed by the Auditor and the Crypto-officer. FMT_MSA.1 (Object Attributes), FMT_MSA.2 (Object Attributes), FMT_MSA.3 (Object Attributes) require controls over the management of object attributes needed to support access control. The user identification and authentication needed to support enforcement of the access control policy is provided by the SFRs satisfying O.User_Authentication and O.Multi-Person_Control.
O.Data_Exchange_Protect	FCS_COP.1 (RSA Enc/Dec), FCS_COP.1 (TDES Enc/Dec), FCS_COP.1 (AES Enc/Dec), FCS_COP.1 (SIGN), FCS_COP.1 (DIGEST), FDP_ACC.1 (TAC), FDP_ACF.1 (TAC), FDP_ETC.1, FDP_ITC.1, FDP_UCT.1, FDP_UIT.1	This combination of SFRs satisfies the objective by requiring that the TOE provide controls over export and import of user data (FDP_ACC.1 (TAC), FDP_ACF.1 (TAC), FDP_ETC.1, FDP_ITC.1, FDP_UCT.1, FDP_UIT.1) plus the cryptographic functions and approved algorithms needed to protect data being exchanged (FCS_COP.1 (RSA Enc/Dec), FCS_COP.1 (TDES Enc/Dec), FCS_COP.1 (AES Enc/Dec), FCS_COP.1 (SIGN), FCS_COP.1 (DIGEST)).
O.Detect_Attack	FPT_PHP.2, FPT_PHP.3	The SFR FPT_PHP.2 implements notification of and FPT_PHP.3 resistance to physical attack. The refinements limit the tamper scenarios to opening the device or removal of a cover. This limitation is reasonable because RE.ENV_Protect_Access requires CSP security measures for physical protection of the TOE.

Objectives	Security Functional Requirements	Rationale
O.Error_Secure	FPT_AMT.1, FPT_FLS.1, FPT_RCV.1, FPT_TST.1, FRU_FLT.1	The SFR FPT_AMT.1 and FPT_TST.1 require tests for error detection and the SFR FPT_FLS.1 requires preservation of a secure state when errors are detected. The TSF shall destroy the plaintext SCP-SCD and other confidential secret and private keys if failures occur. The SFR FPT_RCV.1 requires a mode where the ability to return the TOE to a secure state is provided. FRU_FLT.1 requires that the TOE's data protection continues to be in place in the event of power or data I/O failure.
O.Import_Code	FTP_ITC.1 (FW Update), FCS_CKM.2 (FW Update), FCS_COP.1 (TDES Enc/Dec)	This combination of SFRs satisfies the objective by requiring that the TOE provide the mechanisms and approved algorithms needed to receive a firmware update package from the vendor in a trusted manner and for the TOE to verify the authenticity of the firmware update.
O.Key_Secure	FCS_CKM.1, FCS_CKM.2 (BACKUP), FCS_CKM.3, FCS_CKM.4, FCS_COP.1 (SIGN), FCS_RND.1, FDP_ACC.1 (CRYPTO), FDP_ACF.1 (CRYPTO), FDP_ACC.1 (TAC), FDP_ACF.1 (TAC), FDP_BKP.1, FDP_IFC.1 (CRYPTO), FDP_IFF.4 (CRYPTO), FDP_RIP.1, FDP_SDI.2	The SFRs ensure the cryptographically secure key and key pair generation by FCS_CKM.1 and FCS_RND.1 as well as operation by FCS_COP.1/SIGN according to the list of approved algorithms and parameters. FCS_CKM.3 ensures the security of keys in storage and when accessed by a user. The confidentiality and integrity of the keys will be protected by SFR FDP_RIP.1 and FDP_SDI.2 during internal processing. The SFR FCS_CKM.4 requires secure key destruction to prevent any misuse of keys after their operational life time. FDP_BKP.1 and FCS_CKM.2 (BACKUP) ensure that keys remain secure when they are backed up. The overall key management and operation is under access control of the SFR FDP_ACC.1 (CRYPTO), FDP_ACF.1 (CRYPTO), FDP_ACC.1 (TAC) and FDP_ACF.1 (TAC). The TOE shall protect keys against side-channels by the SFR FDP_IFC.1 (CRYPTO) and FDP_IFF.4 (CRYPTO).
O.Multi-Person_Control	FIA_UAU.5	This SFR satisfies the objective by requiring that the TOE provide a mechanism for multi-person control over access to the TOE's functions.
O.Secure_Init	FPT_AMT.1, FPT_FLS.1, FPT_TST.1	This combination of SFRs satisfies the objective by requiring that the TOE ensure that it is in its initial secure state immediately upon power-up, reset, or after other restart conditions.

Objectives	Security Functional Requirements	Rationale
O.Self_Protect	FPT_RVM.1, FPT_SEP.1	This combination of SFRs satisfies the objective by requiring that the TOE protect its own functions by requiring that enforcement functions are invoked and succeed before allowing operations to proceed and maintaining a separate execution space for its functions.
O.Sign_Secure	FCS_COP.1 (SIGN), FDP_DAU.2, FDP_IFC.1, FDP_IFF.4	The cryptographic security of signatures is implemented by the SFR FCS_COP.1 (SIGN) with reference a list of approved algorithms and parameters. The SFR FDP_IFC.1 (CRYPTO) and FDP_IFF.4 (CRYPTO) requires TSF to prevent illicit information flow about the CSP-SCD through side-channels in the signatures. FDP_DAU.2 supports secure signature by providing evidence that guarantees the authenticity of the signing key and the fact that it was generated in an approved crypto module, plus providing the identity of the guarantor.
O.User_Authentication	FIA_ATD.1, FIA_AFL.1 (SO), FIA_AFL (User), FIA_UID.1, FIA_UAU.1, FIA_UAU.4, FIA_SOS.1, FIA_SOS.2, FIA_USB.1, FMT_MTD.1 (RAD), FMT_MTD.1 (USER_Crypto), FMT_MTD.1 (USER_Audit), FMT_MTD.1 (Login_Failures), FMT_MTD.1 (UAV), FMT_MTD.1 (SOV), FMT_SMF.1, FTP_TRP.1	This combination of SFRs satisfies the objective because the requirements, FIA_ATD.1, FIA_UID.1, FIA_UAU.4 FIA_SOS.1 FIA_SOS.2, FIA_USB.1, provide identification/authentication mechanisms, using randomly generated secrets of specified minimum lengths, and that users are bound to subjects. FIA_AFL.1 (SO), FIA_AFL (User) protect against password guessing attacks and FTP_TRP.1 protects against snooping attacks. The SFRs , FMT_MTD.1 (RAD), FMT_MTD.1 (USER_Crypto), FMT_MTD.1 (USER_Audit), FMT_MTD.1 (Login_Failures), FMT_MTD.1 (UAV), FMT_MTD.1 (SOV), FMT_SMF.1 provide management functions for identification and authentication.
O.User_Data_Protect	FDP_DAU.1, FDP_DAU.2, FDP_RIP.1, FDP_RIP.2, FDP SDI.2	This combination of SFRs satisfies the objective by requiring that the TOE provide mechanisms to protect the confidentiality and integrity of user objects within the TOE and provide the means for the user to verify the integrity of the user object data.
O.ENV_Application	FDP_ACC.1 (CLIENT), FDP_ACF.1 (CLIENT), FDP_UIT.1, FIA_UAU.1 (CLIENT), FIA_UID.1 (CLIENT)	The client application shall implement end-user identification and authentication required by the SFR FIA_UID.1 (CLIENT) and FIA_UAU.1 (CLIENT). It shall implement access control for the DTBS representation sent to the TOE for signing according to the SFR FDP_ACC.1 (CLIENT) and FDP_ACF.1 (CLIENT). Security controls in the TOE environment shall also prevent unauthorised manipulation of data submitted to the TOE as required by SFR FDP_UIT.1.

Objectives	Security Functional Requirements	Rationale
O.ENV_Audit	FAU_SAR.1 (ENV), FAU_STG.1 (ENV)	The audit review of TOE's audit data is implemented in the IT environment by the SFR FAU_SAR.1. Because the TOE implements access control on reading the TOE's audit trail only the SFR FAU_STG.1 (ENV) ensures the availability of the TOE audit trail and prevents the modification of the TOE audit trail outside the TOE.
O.ENV_AuthData	RE.ENV_Personnel	The non-IT requirement, RE.ENV_Personnel, ensures that personnel are aware of their obligations, including the obligation to properly protect their authentication data.
O.ENV_Backup	FCS_CKM.2 (ENV/BACKUP), FCS_COP.1 (ENV/BACKUP_ENC), FCS_COP.1 (ENV/BACKUP_INT), FDP_ACC.1 (ENV/BACKUP), FDP_ACF.1 (ENV/BACKUP), FTP_ITC.1 (ENV/Key Cloning)	This combination security requirements for the IT environment satisfies the objective by ensuring that there is a means provided by the environment to protect the confidentiality of the backup data and detect loss of the integrity of the backup keys, other user data and TSF data needed to restore an operational state after failure when it is transmitted and stored in the TOE environment.
O.ENV_Human_Interface	FIA_UAU.1 (CLIENT), FIA_UID.1 (CLIENT), FTP_TRP.1 (CLIENT)	These SFRs require that the IT environment provide a human interface for identification and authentication that communicates with the TOE via a trusted path.
O.ENV_Outage_Protection	RE.ENV_Outage_Protection	RE.ENV_Outage_Protection requires the CSP to ensure that the power supplied to the TOE is adequately protected against unexpected interruptions and the effects of surges and voltage fluctuations outside the normal operating range of the device and that the TOE is operated in an environment that is provided adequate protection against disasters such as fire and flood.
O.ENV_Personnel	RE.ENV_Personnel	RE.ENV_Personnel implements the definition of the obligations, the services and the roles of the TOE users. The CSP shall inform about their civil, financial and legal responsibilities and train the personnel for their roles.
O.ENV_Protect_Access	RE.ENV_Protect_Access	RE.ENV_Protect_Access requires the CSP to establish physical and organisational security measures against modification of TOE hardware, firmware and software. These measures shall restrict the access to the TOE and protected assets to authorised persons. Note that the TOE itself protects by FPT_PHP.2 and FPT_PHP.3 the confidentiality of the CSP-SCD against physical access because even the CSP personnel do not need to know the CSP-SCD in plaintext.

Objectives	Security Functional Requirements	Rationale
O.ENV_Recovery	RE.ENV_Recovery	RE.ENV_Recovery implements recovery plans and procedures using the TOE TSF according to FDP_BKP.1 and other SFR. It takes recovery in case of detected errors or physical tampering into account.
O.ENV_Secure_Init	RE.ENV_Secure_Init	RE.ENV_Secure_Init implements the definition and application of procedures and controls set-up the TOE for the secure generation of CSPSCD and initialisation of the signature function.
O.ENV_Secure_Oper	RE.ENV_Secure_Oper	RE.ENV_Secure_Oper requires the implementation of such procedures and controls and the observance of the TOE guidance.
O.ENV_Signed_FW_Update	FCS_CKM.1 (ENV/FW Update), FCS_CKM.2 (ENV/FW Update), FCS_COP.1 (ENV/ENC FW Update), FCS_COP.1 (ENV/SIGN FW Update), FTP_ITC.1 (ENV/FW Update)	This combination of SFRs satisfies the objective by requiring that the IT environment provide an application to generate signed and encrypted firmware update packages in a manner that allows the receiving TOE instance to verify its authenticity and decrypt and load the updated firmware.

Table 8-6: Dependency Rationale for Security Functional Requirements

Security Functional Requirement	Dependencies	Rationale
FAU_GEN.1	FPT_STM.1 Reliable time stamps	FPT_STM.1 is not included as an SFR. FAU_GEN.1 2 is refined to state that time stamps are not available in the TOE and sequence numbers are used instead.
FAU_GEN.2	FAU_GEN.1, FIA_UID.1	Met by inclusion of FAU_GEN.1, FIA_UID.1.
FAU_STG.2	FAU_GEN.1	Met by inclusion of FAU_GEN.1
FCS_CKM.1	[FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	Met by inclusion of FCS_COP.1, FCS_CKM.4 and FMT_MSA.2 as SFRs
FCS_CKM.2	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	Met by inclusion of FDP_ITC.1, FCS_CKM.1, FCS_CKM.4 and FMT_MSA.2 as SFRs.
FCS_CKM.3	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	Met by inclusion of FDP_ITC.1 and FCS_CKM.1, FCS_CKM.4 and FMT_MSA.2 as SFRs
FCS_CKM.4	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FMT_MSA.2 Secure security attributes	Met by inclusion of FDP_ITC.1 and FCS_CKM.1 and FMT_MSA.2 as SFRs
FCS_COP.1	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	Met by inclusion of FDP_ITC.1 and FCS_CKM.1, FCS_CKM.4 and FMT_MSA.2 as SFRs
FCS_RND.1	FPT_TST.1	Met by inclusion of FPT_TST.1 – self-tests include test of RNG.
FDP_ACC.1	FDP_ACF.1 Security attribute based access control	Met by inclusion of FDP_ACF.1 for each iteration
FDP_ACF.1	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization	Met by inclusion of FDP_ACC.1 for each iteration and FMT_MSA.3
FDP_BKP.1	[FCS_CKM.1 Cryptographic key generation or FCS_CKM.2 Cryptographic key distribution or FDP_ITC.1 Import of user data without security attributes] FCS_COP.1 Cryptographic operation	Met by inclusion of FCS_CKM.1, FCS_CKM.2 (BACKUP), FCS_COP.1(BACKUP_ENC) and FCS_COP.1(BACKUP_INT)
FDP_DAU.1	No dependencies	No dependencies
FDP_DAU.2	FIA_UID.1	Met by inclusion of FIA_UID.1 as SFR

Security Functional Requirement	Dependencies	Rationale
FDP_ETC.1	FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control	Met by inclusion of FDP_ACC.1
FDP_ITC.1	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_MSA.3 Static attribute initialization	Met by inclusion of FDP_ACC.1 and FMT_MSA.3
FDP_IFC.1	FDP_IFF.1	FDP_IFC.1 is defined without reference to any security attributes.
FDP_IFF.4	FDP_IFC.1, AVA_CCA.1	Met by inclusion of FDP_IFC.1, AVA_CCA.1.
FDP_RIP.1	No dependencies	No dependencies
FDP_RIP.2	No dependencies	No dependencies
FDP SDI.2	No dependencies	No dependencies
FDP_UCT.1	[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	Met by inclusion of FTP_TRP.1 and FDP_ACC.1
FDP_UIT.1	[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	Met by inclusion of FTP_TRP.1 and FDP_ACC.1
FIA_AFL.1	FIA_UAU.1 Timing of authentication	Met by inclusion of FIA_UAU.1
FIA_ATD.1	No dependencies	No dependencies
FIA_SOS.1	No dependencies	No dependencies
FIA_SOS.2	No dependencies	No dependencies
FIA_UAU.1	FIA_UID.1 Timing of identification	Met by inclusion of FIA_UID.1
FIA_UAU.4	No dependencies	No dependencies
FIA_UAU.5	No dependencies	No dependencies
FIA_UID.1	No dependencies	No dependencies
FIA_USB.1	FIA_ATD.1 User attribute definition	Met by inclusion of FIA_ATD.1
FMT_MOF.1	FMT_SMF.1 Specification of management functions, FMT_SMR.1 Security roles	Met by inclusion of FMT_SMF.1 and FMT_SMR.1
FMT_MSA.1	[FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles	Met by inclusion of FDP_ACC.1 and FMT_SMR.1
FMT_MSA.2	ADV_SPM.1 Informal TOE security policy model [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	Met by inclusion of FDP_ACC.1, FMT_MSA.1, FMT_SMR.1 and ADV_SPM.1
FMT_MSA.3	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	Met by inclusion of FMT_MSA.1 and FMT_SMR.1
FMT_MTD.1	FMT_SMR.1 Security roles	Met by inclusion of FMT_SMR.1
FMT_SMF.1	No dependencies	No dependencies
FMT_SMR.1	FIA_UID.1 Timing of identification	Met by inclusion of FIA_UID.1
FPT_AMT.1	No dependencies	No dependencies

Security Functional Requirement	Dependencies	Rationale
FPT_FLS.1	ADV_SPM.1 Informal TOE security policy model	Met by provision of Informal Security Policy Model
FPT_ITC.1	No dependencies	No dependencies
FPT_ITI.1	No dependencies	No dependencies
FPT_PHP.2	FMT_MOF.1	Local user is informed of tamper event by removal/non-availability of TOE. No management of security functions is required.
FPT_PHP.3	No dependencies	No dependencies
FPT_RCV.1	AGD_ADM.1 Administrator guidance ADV_SPM.1 Informal TOE security policy model	Met by inclusion of AGD_ADM.1 and ADV_SPM.1.
FPT_RVM.1	No dependencies	No dependencies
FPT_SEP.1	No dependencies	No dependencies
FPT_TST.1	FPT_AMT.1 Abstract machine testing	Met by inclusion of FPT_AMT.1.
FRU_FLT.1	FPT_FLS.1 Failure with preservation of secure state	Met by inclusion of FPT_FLS.1.
FTP_ITC.1	No dependencies	No dependencies
FTP_TRP.1	No dependencies	No dependencies
SFRs for TOE Environment		
FAU_SAR.1	FAU_GEN.1 Audit data generation	Met because audit records provided by FAU_GEN.1 in TOE.
FAU_STG.1	FAU_GEN.1 Audit data generation	Met because audit records provided by FAU_GEN.1 in TOE.
FCS_CKM.1	[FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	FCS_COP.1, FCS_CKM.4 are included as SFRs for the environment. Because the full specification of the firmware update application system is outside the scope of this ST, dependencies for FMT_MSA.2 are impossible to satisfy and it has therefore not been included as an SFR for the environment.
FCS_CKM.2	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	FCS_CKM.1, FCS_CKM.4 are included as SFRs for the environment. Because the full specification of the firmware update application system is outside the scope of this ST, dependencies for FMT_MSA.2 are impossible to satisfy and it has therefore not been included as an SFR for the environment.

Security Functional Requirement	Dependencies	Rationale
FCS_COP.1	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes	Met by inclusion of FCS_CKM.1, FCS_CKM.4 as SFRs for the environment. Because the full specification of the firmware update application system is outside the scope of this ST, dependencies for FMT_MSA.2 are impossible to satisfy and it has therefore not been included as an SFR for the environment.
FDP_ACC.1	FDP_AFC.1	Met by inclusion of FDP_ACF.1 as a SFR for the environment.
FDP_ACF.1	FDP_ACC.1, FMT_MSA.3	FDP_ACC.1 is included as a SFR for the environment. FMT_MSA.3 is not included because the cryptographic module does not need to specify requirements for management of security attributes of the client application. It is up to the CSP to define which kind of static attribute initialisation of the client application (either permissive or restrictive in nature) ensures that the default values of security attributes are appropriate.
FDP_UIT.1	FDP_ACC.1, FTP_TRP.1	Met by inclusion of FDP_ACC.1, FTP_TRP.1
FIA_UAU.1	FIA_UID.1	Met by inclusion of FIA_UID.1
FIA_UID.1	No dependencies	No dependencies
FTP_ITC.1	No dependencies	No dependencies
FTP_TRP.1	No dependencies	No dependencies

Table 8-7 – Mapping of IT Security Functions to IT Security Requirements and SFRs

IT Security Function	TSS Reference	CC Requirement Title	CC Functional Component
Audit Data Generation	6.3.1	Audit data generation	FAU_GEN.1
		User identity association	FAU_GEN.2
		Guarantees of audit data availability	FAU_STG.2
User Identification and Authentication	6.3.3	Timing of identification	FIA_UID. 1
		Timing of authentication	FIA_UAU.1
		Single-use authentication mechanisms	FIA_UAU.4
		Multiple authentication mechanisms	FIA_UAU.5
		Authentication failure handling	FIA_AFL.1
		User subject binding	FIA_USB.1
Trusted Path – Luna PED	6.3.2	Trusted path	FTP_TRP.1
Authentication data selection	6.3.4	Verification of secrets	FIA_SOS.1
		TSF generation of secrets	FIA_SOS.2
User account data	6.3.5	User attribute definition	FIA_ATD.1
TOE Roles	6.1.3	Security roles	FMT_SMR.1
Access Control	6.3.6	Subset access control	FDP_ACC.1
		Security attribute based access control	FDP_ACF.1
Object Re-Use	6.3.7	Subset residual information protection	FDP_RIP.1
		Full residual information protection	FDP_RIP.2
Data Authentication	6.3.8	Basic data authentication	FDP_DAU.1
		Data authentication with identity of guarantor	FDP_DAU.2
Key Pair Integrity Checking	6.3.9	Stored data integrity monitoring and action	FDP_SDI.2
Key Export and Import Protection	6.3.10	Imported user data without security attributes	FDP_ITC.1
		Exported user data without security attributes	FDP_ETC.1
Cryptographic Material Management	6.3.11	Cryptographic key generation	FCS_CKM.1
	6.3.11.1	Cryptographic key access	FCS_CKM.3
	6.3.11	Cryptographic key destruction	FCS_CKM.4
	6.3.11	Subset information flow control	FDP_IFC.1
Cryptography	6.3.12	Cryptographic operation	FCS_COP.1
		Quality metrics for random numbers	FCS_RND.1
		Subset information flow control	FDP_IFC.1
Data Exchange	6.3.13	Basic data exchange confidentiality	FDP_UCT.1
		Data exchange integrity	FDP_UIT.1
		Inter-TSF trusted channel	FTP_ITC.1
Specification of Security Management Functions	6.3.14	Specification of management functions	FMT_SMF.1
Security Function Management	6.3.15	Management of security functions behaviour	FMT_MOF.1

IT Security Function	TSS Reference	CC Requirement Title	CC Functional Component
Security Data Management	6.3.16	Management of security attributes	FMT_MSA.1
		Secure security attributes	FMT_MSA.2
		Static attribute initialization	FMT_MSA.3
		Management of TSF data	FMT_MTD.1
Memory and Firmware Integrity Check	6.3.17.1	Abstract machine testing	FPT_AMT.1
Self-Tests	6.3.17.2	TSF testing	FPT_TST.1
Prevention of By-pass and Separate Execution Domain	6.3.17.3	Non-bypassibility of the TSP	FPT_RVM.1
		TSF domain separation	FPT_SEP.1
Preservation of Secure State	6.3.17.4	Failure with preservation of secure state	FPT_FLS.1
Preservation of Secure State	6.3.17.4	Manual recovery	FPT_RCV.1
Firmware Loading and Firmware Update	6.3.17.5	Cryptographic key distribution	FCS_CKM.2
		Inter-TSF trusted channel	FTP_ITC.1
Cloning	6.3.18	Cryptographic key distribution	FCS_CKM.2
		TSF generation of secrets	FIA_SOS.2
		Inter-TSF trusted channel	FTP_ITC.1
Physical Self-Protection	6.3.19	Passive detection of physical attack	FPT_PHP.2
		Resistance to physical attack	FPT_PHP.3
Failure handling	6.3.20	Degraded fault tolerance	FRU_FLT.1
Backup and Recovery	6.3.21	Cryptographic key distribution	FCS_CKM.2
		Cryptographic operation	FCS_COP.1
		Exported user data without security attributes	FDP_ETC.1
		Backup	FDP_BKP.1
		Subset information flow control	FDP_IFC.1
		Partial elimination of illicit information flows	FDP_IFF.4
		Inter-TSF confidentiality during transmission	FPT_ITC.1
		Inter-TSF detection of modification	FPT_ITI.1
		Inter-TSF trusted channel	FTP_ITC.1

Table 8-8 – Mapping of Security Functional Requirements to IT Security Functions

CC Requirement Title	CC Functional Component	ST Reference	IT Security Function	TSS Reference	Rationale
Audit data generation	FAU_GEN.1	5.1.1.1	Audit Data Generation	6.3.1	The security function satisfies the SFR by providing audit data, in the form of commands and responses with associated sequence numbers, that is accessible by an audit review application in the environment.
User identity association	FAU_GEN.2	5.1.1.2	Audit Data Generation	6.3.1	The security function satisfies the SFR by providing audit data that can be associated with the user responsible for invoking a command.
Guarantees of audit data availability	FAU_STG.2	5.1.1.3	Audit Data Generation	6.3.1	The security function satisfies the SFR by ensuring that 60kB of audit data is always available to be read by the application in the environment.
Cryptographic key generation	FCS_CKM.1	5.1.2.1	Cryptographic Material Management	6.3.11	The security function satisfies the SFR by providing mechanisms for the generation of RSA, DSA and TDES keys of the specified lengths in accordance with the appropriate standards.
Cryptographic key distribution	FCS_CKM.2	5.1.2.2	Backup and Recovery	6.3.21	The security function satisfies the SFR by providing a secure means to derive a key, as part of the cloning protocol, that is used to encrypt the backup data when it is transferred from the TOE to a backup token.
Cryptographic key distribution	FCS_CKM.2	5.2.1.1	Firmware Loading and Firmware Update	6.3.17.5	The security function satisfies the SFR by implementing a symmetric key distribution protocol for the protection of firmware upgrade packages.
Cryptographic key access	FCS_CKM.3	5.2.1.2	Key Storage and Access Protection	6.3.11.1	The security function satisfies the SFR by providing mechanisms for key storage and access in accordance with the PKCS #11 standard.
Cryptographic key destruction	FCS_CKM.4	5.1.2.3	Cryptographic Material Management	6.3.11	The security function satisfies the SFR by providing mechanisms that destroys keys in accordance with the PKCS #11 and FIPS 140-2 Level 3 standards.
Cryptographic operation	FCS_COP.1	5.1.2.4, 5.1.2.5, 5.1.2.6 & 5.2.1.3, 5.2.1.4, 5.2.1.5, 5.2.1.6	Cryptography	6.3.12	The security function satisfies the SFR by providing mechanisms that implement the specified set of cryptographic algorithms in accordance with the appropriate standards.
Quality metrics for random numbers	FCS_RND.1	5.1.2.7	Cryptography	6.3.12	The security function satisfies the SFR by providing a Random Number Generator that conforms to a FIPS 140-2 Level 3 validated Random Number Generator (RNG).
Subset access control	FDP_ACC.1	5.1.3.1, 5.1.3.2, 5.1.3.3, 5.2.2.1	Access Control	6.3.6	The security function satisfies the SFR by enforcing the Token Access Control policy on subjects (sessions), objects and a set of controlled operations.

CC Requirement Title	CC Functional Component	ST Reference	IT Security Function	TSS Reference	Rationale
Security attribute based access control	FDP_ACF.1	5.1.3.4, 5.1.3.5, 5.1.3.6, 5.2.2.2	Access Control	6.3.6	The security function satisfies the SFR by enforcing the TAC Policy based on the specified sets of subject and object attributes. The access rules for subjects, objects and operations are as given by table 5-2.
Backup and recovery	FDP_BKP.1	5.1.3.7	Backup and Recovery	6.3.21	The security function satisfies the SFR by providing backup and recovery functions that can be invoked on demand by an authorised user and that protects the confidentiality of the backup data by encryption and the integrity with SHA-1 checksums. The backup data is sufficient to re-create the stored state of one instance of the TOE on a second instance of the TOE for recovery purposes.
Basic data authentication	FDP_DAU.1	5.2.2.3	Data Authentication	6.3.8	The security function satisfies the SFR by providing a SHA-1 fingerprint for stored objects that can be queried by the user in order to validate the integrity of the object.
Data authentication with identity of guarantor	FDP_DAU.2	5.2.2.4	Data Authentication	6.3.8	The security function satisfies the SFR by providing a Public Key Confirmation mechanism that can be used to guarantee the validity of data objects and verify the identity of the originator who performed the digital signature.
Export of user data without security attributes	FDP_ETC.1	5.1.3.8	Key Export and Import Protection, Backup and Recovery	6.3.10, 6.3.21	The security function satisfies the SFR by enforcing the TAC when data is exported through a Wrap operation. Objects are exported without security-related attributes.
Subset information flow control	FDP_IFC.1	5.1.3.9 5.1.3.10	Backup and Recovery, Cryptographic Material Management, Cryptography	6.3.21, 6.3.11, 6.3.12	These security functions satisfy the SFR by ensuring that there is no illicit information flow associated with backup recovery, key management and cryptographic operations that could compromise the CSP_SCD.
Partial elimination of illicit information flows	FDP_IFF.4	5.1.3.11, 5.1.3.12	Backup and Recovery, Cryptographic Material Management, Cryptography	6.3.21, 6.3.11, 6.3.12	These security functions satisfy the SFR by ensuring that there is no illicit information flow associated with backup recovery, key management and cryptographic operations that could compromise the CSP_SCD.
Imported user data without security attributes	FDP_ITC.1	5.2.2.5	Key Export and Import Protection	6.3.10	The security function satisfies the SFR by enforcing the TAC when data is imported through an Unwrap operation. The TSF ignores any security-related attributes that may have been associated with the imported object and sets the object's attributes to the appropriate values for its type and, in particular, the CKA_SENSITIVE attribute is always set.
Subset residual information protection	FDP_RIP.1	5.1.3.13	Object Reuse	6.3.7	The security function satisfies the SFR by ensuring that the information content of private keys and secret keys is made unavailable upon the de-allocation of the resource associated with a key.

CC Requirement Title	CC Functional Component	ST Reference	IT Security Function	TSS Reference	Rationale
Full residual information protection	FDP_RIP.2	5.2.2.6	Object Reuse	6.3.7	The security function satisfies the SFR by ensuring that the information content of resources is made unavailable when the resource is re-allocated.
Stored data monitoring and action	FDP_SDI.2	5.1.3.14	Key Pair Integrity Checking	6.3.9	The security function satisfies the SFR by providing a means to check the integrity of a private key object before output of a digital signature by verifying the digital signature created with the private key using the corresponding public key object for signature verification.
Basic data exchange confidentiality	FDP_UCT.1	5.2.2.7	Data Exchange	6.3.13	The security function satisfies the SFR by providing encryption mechanisms and a logical trusted channel to protect transmitted and received objects from unauthorised disclosure using the Wrap and Unwrap operations.
Data exchange integrity	FDP_UIT.1	5.2.2.8	Data Exchange	6.3.13	The security function satisfies the SFR by providing digital signature mechanisms to protect transmitted and received objects from modification and insertion errors.
Authentication failure handling	FIA_AFL.1	5.1.4.1 & 5.1.4.2	User Identification and Authentication	6.3.3	The security function satisfies the SFR by detecting when the maximum number of login failures occur (3 for SO, set in the TPV for Token User) and performing on of the following: Zeroize the device in the case of SO authentication failure Remove the user and zeroize the user's memory space, if a Token User authentication failure.
User attribute definition	FIA_ATD.1	5.1.4.3	User Account Data	6.3.5	The security function satisfies the SFR by maintaining the required list of security attributes within the UAV for each Token User.
Verification of secrets	FIA_SOS.1	5.1.4.4	Authentication Data Selection	6.3.4	The security function satisfies the SFR by requiring that PIN values meet the minimum and maximum length constraints established by the SO via the TPV.
TSF generation of secrets	FIA_SOS.2	5.2.3.1	Authentication Data Selection	6.3.4	The security functions satisfy the SFR by generating random authentication data of the required lengths for each of the functions for which they are required.
Timing of authentication	FIA_UAU.1	5.1.4.5	User Identification and Authentication	6.3.3	The security function satisfies the SFR by allowing a user to perform a specified set of actions before authentication and by requiring the user to be successfully authenticated before allowing the user to perform any other actions on the module.
Single-use authentication mechanisms	FIA_UAU.4	5.2.3.2	User Identification and Authentication	6.3.3	The security function satisfies the SFR by implementing the challenge-response scheme as a single-use authentication mechanism.

CC Requirement Title	CC Functional Component	ST Reference	IT Security Function	TSS Reference	Rationale
Multiple authentication mechanisms	FIA_UAU.5	5.2.3.3	User Identification and Authentication	6.3.3	The security function satisfies the SFR by providing multiple authentication mechanisms including PED key, PED key and PED PIN and M of N.
Timing of identification	FIA_UID. 1	5.1.4.6	User Identification and Authentication	6.3.3	The security function satisfies the SFR by allowing a user to perform a specified set of actions before identification and by requiring the user to be successfully identified before allowing the user to perform any other actions on the module.
User subject binding	FIA_USB.1	5.2.3.4	User Identification and Authentication	6.3.3	The security function satisfies the SFR by specifying that the user identity be bound to the subject (session) acting on behalf of the user by including the UAV data within the session state.
Management of security functions behaviour	FMT_MOF.1	5.2.4.1	Security Function Management	6.3.15	The security function satisfies the SFR by restricting the ability to perform the specified security management operations to the SO role.
Management of security attributes	FMT_MSA.1	5.1.5.1, 5.1.5.2, 5.2.4.2	Security Data Management	6.3.16	The security function satisfies the SFR by enforcing the TAC Policy to restrict the ability to manipulate user and object security attributes as specified.
Secure security attributes	FMT_MSA.2	5.1.5.3, 5.2.4.3	Security Data Management	6.3.16	The security function satisfies the SFR by ensuring that only secure values are accepted for security attributes.
Static attribute initialization	FMT_MSA.3	5.1.5.4, 5.2.4.4	Security Data Management	6.3.16	The security function satisfies the SFR by requiring restrictive values for security attributes that cannot be changed based on the capability and policy settings.
Management of TSF data	FMT_MTD.1	5.1.5.5, 5.1.5.6, 5.1.5.7, 5.1.5.8, 5.1.5.9, 5.2.4.5, 5.2.4.6, 5.2.4.7	Security Data Management	6.3.16	The security function satisfies the SFR by enforcing the TAC Policy to restrict the ability to manipulate the policy settings to the SO.
Specification of management functions	FMT_SMF.1	5.1.5.10, 5.2.4.8	Specification of Security Management Functions	6.3.14	The security function satisfies the SFR by specifying the security management functions that may be performed.
Security roles	FMT_SMR.1	5.1.5.11	TOE Roles	6.1.3	The security function satisfies the SFR by specifying the security roles that are implemented by the TOE – Security Officer, Crypto Officer and Crypto User.
Abstract machine testing	FPT_AMT.1	5.1.6.1	Memory and Firmware Integrity Check	6.3.17.1	The security function satisfies the SFR by running a suite of tests at startup and upon user request to verify the correct operation of the security-relevant aspects of the underlying module hardware.
Failure with preservation of secure state	FPT_FLS.1	5.1.6.2	Preservation of Secure State	6.3.17.4	The security function satisfies the SFR by preserving the module in a secure state when the specified failure conditions occur.
Inter-TSF confidentiality during transmission	FPT_ITC.1	5.1.6.3	Backup and Recovery	6.3.21	The security function satisfies the SFR by encrypting the transmitted data according to the cloning protocol.

CC Requirement Title	CC Functional Component	ST Reference	IT Security Function	TSS Reference	Rationale
Inter-TSF detection of modification	FPT_ITI.1	5.1.6.4	Backup and Recovery	6.3.21	The security function satisfies the SFR by providing the same data authentication mechanism (FDP_DAU.1) at both the TOE and backup token.
Notification of physical attack	FPT_PHP.2	5.1.6.5	Physical Self-Protection	6.3.19	The security function satisfies the SFR by implementing physical security mechanisms that provide unambiguous evidence of physical tampering and the ability to determine whether physical tampering with security-relevant devices has occurred.
Resistance to physical attack	FPT_PHP.3	5.1.6.6	Physical Self-Protection	6.3.19	The security function satisfies the SFR by implementing physical security mechanisms that resist tampering such that opening the module's enclosure results in plaintext key material and other sensitive data being erased from the module.
Manual recovery	FPT_RCV.1	5.1.6.7	Preservation of Secure State	6.3.17.4	The security function satisfies the SFR by ensuring that the module maintains its secure state in the event of failure or service discontinuity and can be returned to operation in its secure state once the failure has been resolved.
Non-bypassibility of the TSP	FPT_RVM.1	5.2.5.1	Prevention of By-pass and Separate Execution Domain	6.3.17.3	The security function satisfies the SFR by ensuring that TSP enforcement functions are invoked and succeed before each function within the module firmware is allowed to proceed.
TSF domain separation	FPT_SEP.1	5.2.5.2	Prevention of By-pass and Separate Execution Domain	6.3.17.3	The security function satisfies the SFR by maintaining a separate domain for the execution of the TOE security functions and by separating subject domains by maintaining cryptographic separation of user data, by allowing only one logged in user to be active on the module and by allowing a single thread of execution on the module.
TSF testing	FPT_TST.1	5.1.6.8	Self-Tests	6.3.17.2	The security function satisfies the SFR by providing a suite of self-tests to verify the correct operation of the security functions on start-up and at the request of an authorised user.

CC Requirement Title	CC Functional Component	ST Reference	IT Security Function	TSS Reference	Rationale
Degraded fault tolerance	FRU_FLT.1	5.2.6.1	Failure Handling	6.3.20	The security function satisfies the SFR by ensuring that the user data protection capabilities are maintained when power failures or data I/O failures occur. The module maintains all Sensitive permanent objects in an encrypted state and, therefore, such failures cannot affect the protection of permanent objects. Volatile objects are wiped from memory when power to the module is lost. Data I/O failures result in suspension of user operations on the module, but data protection capabilities are maintained and the module will return to operation once data I/O is restored in the same state it was prior to the failure.
Inter-TSF trusted channel	FTP_ITC.1	5.2.7.1, 5.2.7.2	Firmware Loading and Firmware Update, Cloning	6.3.17.5, 6.3.18	The security functions satisfy the SFR by: providing a logical trusted channel between a customer module and a separate module containing a firmware image to initially load the customer module or update the firmware on the customer module, providing a logical trusted channel between a customer module and a remote trusted product (backup token) for the purpose of protected key backup and recovery.
Trusted path	FTP_TRP.1	5.1.7.1	Trusted Path – Luna PED	6.3.2	The security function satisfies the SFR by requiring the use of a logically distinct trusted path via the PED and dedicated serial port.

Table 8-9 – Assurance Measures

Assurance Measures	Document Title	Document Description
Configuration Management	Configuration Management Manual for Luna Product	This document describes the CM procedure for Luna product design, development, testing, release and manufacture and maintenance.
	Configuration Management Plan for Luna® SA 4.5.1 with Firmware 4.8.7	This document defines CM procedures to be used during development of the Luna® PCI.
	SafeNet Engineering Change Request Procedure	This procedure establishes the requirements for requesting action that may initiate a release or change to released hardware, software, or controlled documentation during the various product life cycles.
	SafeNet Engineering Change Procedure	This procedure establishes the requirements for documenting and approving Engineering Change Notices (ECNs) to release new items as well as for released hardware, software, or controlled documentation during the various product life cycles.
	Luna Product Identification and Traceability (Sections 2 to 4)	This document describes product identification and marking on all Luna products and specifies the process of maintaining product information for traceability purposes.
	Luna Product Development Access Controls	This document defines the high-level access controls in place to develop Luna products.
	Luna Critical Security Material Handling for Contract Manufacturers	This document describes the various administrative procedures and security policies for handling critical key material that is used during the manufacture of Luna® devices.
	Luna Product Process Flow for Contract Manufacturers	The checklist is intended to show the process flow of Luna finished goods through the Contract Manufacturer's facility.
	Luna Board Level Development Process	This document describes the design procedure for board level development of Luna products. This document does not cover mechanical or system level design procedures.
	Luna Products Life Cycle – Assuring Integrity	This document provides an overview of the procedures and steps taken to assure the integrity of Luna products from development through to delivery.
	Statement of Work (SOW) for Luna SA Release 4.0.5	This document defines the statement of work and product requirements for the release of Luna® SA 4.0.5.
	Statement of Work (SOW) for Luna SA Release 4.5.1	This document defines the statement of work and product requirements for the release of Luna® SA 4.5.1.

Assurance Measures	Document Title	Document Description
Delivery and operation documentation	Luna® SA Checkoff/Quick Start Guide v4.5	Quick Start Guide applicable to Luna® SA Release v4.5 and v4.5.1. Describes the procedures for secure initialization of the product. The Content Sheet identifies the TOE components that the customer should expect to find in the delivered product.
	Luna Critical Security Material Handling for Contract Manufacturers	This document describes the various administrative procedures and security policies for handling critical key material that is used during the manufacture of Luna® devices.
	Luna Product Process Flow for Contract Manufacturers	The checklist is intended to show the process flow of Luna finished goods through the Contract Manufacturer's facility.
Functional Specification – fully defined external interfaces	Luna® Functional Specification	Presents a high-level overview of the architecture of the Luna® PCI, its cryptographic capabilities and its security features with a full definition of all relevant external interfaces.
	PKCS #11: Cryptographic Token Interface Standard, V 2.20	This document is offered by RSA Laboratories to developers of computer systems employing public-key and related technology.
	Extensions to PKCS#11, Cryptographic Token Interface Standard	Describes a set of extensions to the standard application programming interface (API), called Cryptoki. Specifies the data types and functions available to an application requiring cryptographic services using the ANSI C programming language.
Security enforcing high-level design	Luna® High-Level Design	Provides an overview of the cryptographic module's firmware architecture and a foundation for further design documents that address each module in more detail.
	Luna® Interface Control Document (ICD)	Defines the command set and associated parameters that are used to communicate to the Luna® cryptographic module.
Descriptive low-level design	Luna Memory Management Subsystem Design	Identifies the set of requirements met by the Luna® Memory Management Subsystem Design and provides detailed notes on its implementation.
	Luna® Session Manager Subsystem Design	Presents the Luna® Session Manager Subsystem capabilities and describes the implementation, internal interface and external command processing details.
	Luna Object Management Subsystem Design	Identifies the set of requirements to be met by the Luna® Object Management Subsystem Design and provides detailed notes on its design and implementation.
	Luna® Interface Control Document (ICD)	Defines the command set and associated parameters that are used to communicate to the Luna® module.

Assurance Measures	Document Title	Document Description
	Luna® Boot Block Subsystem Design	Identifies the set of requirements met by the Luna® boot block and provides detailed notes on its implementation.
	Luna® Communication Subsystem Design	Identifies the set of requirements to be met by the Luna® Communication Subsystem design and provides detailed notes on its implementation.
	Luna® Param Subsystem Design	Describes the Luna® Param Subsystem Design implementation and defines the internal interface.
	Luna® Cryptographic Module Self Tests	Identifies the set of requirements to be met by the Luna® PCI self tests and provides detailed notes on their implementation.
	Luna® PCI User Subsystem Design	Design and validation document for Luna® firmware user subsystem design.
	Luna Cryptographic Algorithms Subsystem Design	Describes the generation, distribution, loading, storing, use, updating and destruction of the cryptographic material – keys and vectors – necessary for all of the cryptographic operations performed by the cryptographic module.
	Luna Key Cloning Protocol	Describes the key cloning protocol.
	Luna® Key Management Subsystem Design	Describes the generation, distribution, loading, storing, use, updating and destruction of the cryptographic material – keys and vectors – necessary for all of the cryptographic operations performed by the cryptographic module.
	Luna M of N Activation Protocol	Describes the M of N activation capability implemented on the Luna® cryptographic modules.
	Luna® Main Subsystem Design	This document presents the Luna® Main Subsystem low-level design in terms of subordinate modules, their roles within the subsystem and their interactions to provide the Main Subsystem services.
	Luna® Random Number Generation (RNG) process	This document describes the Random Number Generation (RNG) process used to generate the random bits required by the cryptographic functions running on the Luna® family of cryptographic hardware modules hereinafter referred to as Luna® modules.
	Luna® Firmware (F/W) Update High Level Design	Defines the process used to perform a secure update to the firmware on cryptographic modules in the field. The firmware update process maintains all user information that exists on the cryptographic module prior to the update.

Assurance Measures	Document Title	Document Description
	Luna® Serial Communication Protocol for Luna® PCI	This document describes the Luna® Serial Communication Protocol (SCP) used to transfer data over the serial communication port interface defined for the Luna® PCI cryptographic module
	Luna® PCI Key Card HW Assembly Drawings	
	Top Assembly, Luna® SA Key Card (Luna® PCI) Drawings	
Implementation of the TSF		Firmware code, hardware schematics, FPGA code to be provided in accordance with evaluator's request.
Informal correspondence demonstration		Correspondence mappings for TSS to Functional Specification, Functional Specification to HLD and HLD to LLD are provided as part of the appropriate documents.
Informal TOE security policy model	Luna® PCI Informal Security Policy Model	The Informal Security Policy Model describes the security behaviour of the Luna® PCI.
Guidance documents	Guidance documents provided with the TOE are primarily intended as Administrator guidance. The administration functions are normally carried out by the Security Officer, or possibly a designated User, using the CLI software as the interface. In most cases, these functions will be performed very infrequently. User guidance documents are not provided because the normal user of the TOE is an application program making function calls to the TOE via the PKCS #11 Cryptographic API. Direct access to the TOE's functions by a human user only occurs in the course of performing administration functions.	
	Luna® SA Checkoff/Quick Start Guide v4.5	Quick Start Guide applicable to Luna® SA Release v4.5 and v4.5.1.
	Luna SA v4.5 Online Help	The Online Help system provides the detailed Administrator/User guidance for the operation of the product (applicable to Luna® SA Release v4.5 and v4.5.1).
	Luna SA v4.5.1 Customer Release Notes (CRN)	The Release Notes identify major issues from previous releases that have been fixed in the current release and any outstanding known issues.
	Technical Support Service Note for Luna SA 4.0.5	
Life cycle support	Luna Software Development Process	This document defines the high-level software development process used for Luna product.
	Luna Product Identification and Traceability	This document describes product identification and marking on all Luna products and specifies the process of maintaining product information for traceability purpose
	Corporate Encryption Policy	Policies regarding protection of confidential corporate data.

Assurance Measures	Document Title	Document Description
	Luna Board Level Development Process	This document describes the design procedure for board level development of Luna product. This document does not cover mechanical or system level design procedures.
	Return Material Authorization Process	RMA Process for Sales, Service and Logistics
	Luna Life Cycle – Assuring Integrity	This document provides an overview of the procedures and steps taken to assure the integrity of Luna products from development through to delivery.
	Luna Development Tools	This document identifies and explains the use of the various tools used within Luna development environment.
	Luna Product Development Access Controls	This document defines the high-level access controls in place to develop Luna products.
	Luna Critical Security Material Handling for Contract Manufacturers	This document describes the various administrative procedures and security policies for handling critical key material that is used during the manufacture of Luna® devices.
	Luna Product Process Flow for Contract Manufacturers	The checklist is intended to show the process flow of Luna finished goods through the Contract Manufacturer's facility.
	SafeNet Canada Security Policies	This document presents the security policies to be followed by all SafeNet Canada full-time, part-time and contract employees.
	Luna Problem Reporting Process	This document discusses the problem reporting process employed for Software, Firmware and Hardware related problems.
Developer Tests	SafeNet PED Test Description	This document presents the detailed procedures to test SafeNet Canada's PED. The procedures verify that the product features are ready for release to customers.
	ScriptHelp.txt	Help text describing the use of the Scripter tool.
	List of Scripts Used in Engineering Testing and run by Scripter	List of Scripts Used in PV Testing and run by Scripter. Demonstrates tests coverage.
	Cloning Scripts	Script used by PV to test the cloning functionality. Demonstrates tests coverage
	Engineering Test Report for Luna SA 4.5.1	This document provides a view of testing performed on Luna SA 4.5.1 by the Engineering Test team and its readiness to be released to the Quality Assurance (QA) team.
	Engineering Test Report for Luna SA 4.0.5	This document provides a view of testing performed on Luna SA 4.0.5 by the Engineering Test team and its readiness to be released to the Quality Assurance (QA) team.

Assurance Measures	Document Title	Document Description
	SVT Test Report for Luna SA 4.5.1	This document provides the Release Authorization and Test Cases executed for the Luna SA 4.5,1 patch release.
	SVT Test Report for Luna SA 4.0.5	This document provides the Release Authorization and Test Cases executed for the Luna SA 4.0.5 release.
	Testing Coverage and Depth Analysis	This document has been prepared for the Common Criteria evaluation of the Luna® PCI (K5) and describes the functional coverage of the test plan and the depth of testing related to the high-level design interfaces.
	Independent testing - sample	To be performed by evaluator.
Vulnerability assessment	Strength of Function Analysis	This document presents the analysis of the Strength of Function for the permutational and probablisitic mechanisms employed within the Luna® PCI to implement several of its security functions.
	Covert channel Analysis	This document presents the analysis of potential cover channels.
	Misuse Analysis	This document presents the misuse analysis of the Luna® PCI Common Criteria evaluation.

APPENDIX A – REFERENCES

Reference Number	Document Number	Revision	Author	Title
[1]	ISO/IEC 15408-1	V2.3		Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and General Model
[2]	ISO/IEC 15408-2	V2.3		Common Criteria for Information Technology Security Evaluation – Part 2: Security Functional Requirements
[3]	ISO/IEC 15408-3	V2.3		Common Criteria for Information Technology Security Evaluation – Part 3: Security Assurance Requirements
[4]	FIPS 140-2	December 2002	National Institute of Standards and Technology	Security Requirements for Cryptographic Modules
[5]		Version 2.20, June 2004	RSA Laboratories	PKCS #11: Cryptographic Token Interface Standard
[6]		Version 2.1, June 2002	RSA Laboratories	PKCS #1: RSA Cryptography Standard
[7]	CR-2384		SafeNet Canada	Overview of Documentation Required to Support Luna® PCI Common Criteria Evaluation
[8]	CR-3127		SafeNet Canada	Luna Interface Control Document (ICD)