Cifrador Personal del Combatiente

TZ Security Target.
Version: 008

Publication date: Febrary 28, 2020

© TECNOBIT
CIF: B82193210
C/ Marie Curie, 17-19
28521, Rivas-Vaciamadrid, Madrid

DOCUMENT SECURITY CLASSIFICATION:
CONFIDENTIAL TECNOBIT

This document is intended for the use of the recipient only, and for communication to such persons as may be required to be acquainted with its contents in the course of their duties.

The recipient, and any other person to whom the recipient has communicated the contents of this document, shall treat this document in accordance with the requirements of confidential agreement established with TECNOBIT s.l.

Any person other than the authorized holder upon obtaining possession of this document, by finding or otherwise, should forward it by registered post, together with his name and address, in a sealed envelope to:

Chief of Security
TECNOBIT,S.L.
Fudre , 18
13300 Valdepeñas
SPAIN

Personal data protection

Any personal data included in the Contract shall be processed pursuant to Regulation (EC) No 45/2001 on the protection of individuals with regard to the processing of personal data by the Community institutions and bodies and on the free movement of such data. Such data shall be processed solely for the purposes of the performance, management and monitoring of the Contract by *Tecnobit* acting as data controller without prejudice to possible transmission to the bodies charged with monitoring or inspection task in application of Union law.

SYMBOLS AND ABBREVIATED TERMS

A.XXX	Assumption
CC	Common Criteria
CSP	Critical Security Parameter
EAL	Evaluation Assurance Level
IT	Information Technology
O.XXX	Security objective for the TOE
OE.XXX	Security objective for the TOE environment
OS	Operative System
OSP	Organisational Security Policy
PP	Protection Profile
SAR	Security assurance requirement
SFR	Security functional requirement
ST	Security target
T.XXX	Threat
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE security functionality
TSFI	TSF Interface
TSS	TOE Summary Specification

DOCUMENT REFERENCES

[CC31R5P1]	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, Part 1: Introduction and general model
[CC31R5P2]	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, Part 2: Security functional components
[CC31R5P3]	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, Part 3: Security assurance components
[CEM31R5]	Common Criteria Evaluation methodology, Version 3.1, Revision 5
[SCIP108]	Reference Module 108 - Universal Multipoint PPK Key Material Format and Fill rev 1.0
[SCIP305]	Reference Module 305 - Universal Multipoint PPK Processing rev 1.0
[SCIP423]	Reference Module 423 - Universal Fixed Filler Generation rev 1.0
[SCIP442]	Reference Module 442 - Multipoint Cryptographic Verification rev 1.0
[SCIP601]	Reference Module 601 - AES-256 Encryption Algorithm rev 1.0
[SCIP210]	SCIP-210_3.6
[SCIP233.401]	SCIP-233.401-Revision1.3-Application-State-Vector-Processing
[SCIP233.501]	SCIP-233.501-Revision1.3-SecureMELP(e)Voice
[SCIP233]	SCIP-233-Cryptography-Specification-Main-Module-Revision1.1
[RFC4301]	Security Architecture for the Internet Protocol, December 2005

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

1 ST INTRODUCTION

This section identifies the Security Target (ST) and Target of Evaluation (TOE). It also includes the TOE summary and the TOE description. The TOE in this ST is TZ (version 2.4). The TOE is being evaluated as a cryptographic module.

The Security Target contains the following sections:

- *Chapter 1* of this document provides identifying information for the ST and TOE as well as the TOE overview, its associated TOE Type, and a brief description of the TOE, including the physical and logical boundaries.
- *Chapter 2* describes the conformance claims made by this ST.
- *Chapter 3* describes the threats, assumptions, and organizational security policies that apply to the TOE.
- *Chapter 4* describes the security objectives for the TOE as well as the security objectives for the operational environment.
- *Chapter 5* defines extended Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs).
- *Chapter 6* describes the SFRs that are to be implemented by the TSF and the SARs that will be used to evaluate the TOE.
- *Chapter 7* provides the TOE Summary Specification, which describes how the SFRs that are defined for the TOE are implemented by the TSF.

1.1 ST Reference

ST Title: TZ Security Target
ST Version: Version 008
ST Date: Febrary 28, 2020
ST Author: Tecnobit, grupo Oesía
Evaluation lab: Appplus+ laboratories

1.2 TOE Reference

TOE Identification: TZ
TOE Version: 2.4
TOE Developer: Tecnobit, grupo Oesia

1.3 TOE Overview

The Target of Evaluation (TOE) is a cryptographic module that secures communications between different deployed units for a mission and it also implements endorsed cryptographic security functions to protect the confidentiality of user data according to a security policy of an IT system.

The TOE uses, manages and protects the cryptographic keys and missions for these endorsed cryptographic security functions.

A mission consists in a set of TOE devices, configured within a data transmission mode and how that transmission is cryptographically protected.

Cryptographic keys involved in TOE communication provide integrity and confidentiality protection during data transmission.

TOE protects communication data as follows:

- Data: Information is secured following the IPSec protocol according to [RFC4301]. Both, transport and tunnel IPSec modes are able to be configured.

TOE is composed by two physical isolated zones: the red one encrypts and decrypts data sent or to be received by the end user; and the black one manages the delivery of these protected data to one or several endpoints through communication channel.

Devices used by the end user such as computers, which create and process clear data, are connected to the red zone of the TOE.

The media devices that transmit the protected data, such us Tactical IP radios (HARRIS 7800, PR4G, Spearnet, etc.) or network devices such as switches, are connected to the black zone of the TOE.

TOE has two working configurations managed by the role that the user using TOE owns. These roles can be: operator (S.OPERATOR) or management role (S.MANAGEMENT).
When the user using the TOE is an operator role, TOE is able to transmit data in several modes for protecting data information:

- Unicast: one target communication
- Multicast: several targets communication

Management role is capable of getting audit files and to update software, cryptographic keys and missions.

1.3.1 TOE Type

TZ is a portable, light and very small size cryptographic module for protecting data communications.

1.3.2 Required non-TOE hardware/software/firmware.

1.3.2.1 Software identification

A Management Centre (CMAP) is required to support:

- Audit functionalities.

- To create and update cryptographic keys, missions, and software updates and loading them into TZ devices.

CMAP is executed in an isolated computer located in a secured area.

1.3.2.2 Hardware identification

TZ needs a micro-SD card with encrypted information.

The information stored in the micro-SD card are:

- Mission data, containing how the communications between TZ devices integrating the mission, are achieved.
- Cryptographic keys to use in the mission.

1.3.2.3 Components and Applications in the Operational Environment

The following section describes components and applications in the operational environment that the TOE relies upon to work properly.

The operational environment on which the TOE is being installed currently is called CIFPECOM.

CIFPECOM environment is compound by the following components:

1. TZ
2. Micro-SD card
3. Operation service infrastructure.

TZ is the TOE, whose main aim is to protect the data interchanged between different units. All traffic transmitted from and to the TZ is encrypted using:

- IPSec protocol according to [RFC4301] being possible to configure it in both, transport and tunnel IPSec modes.

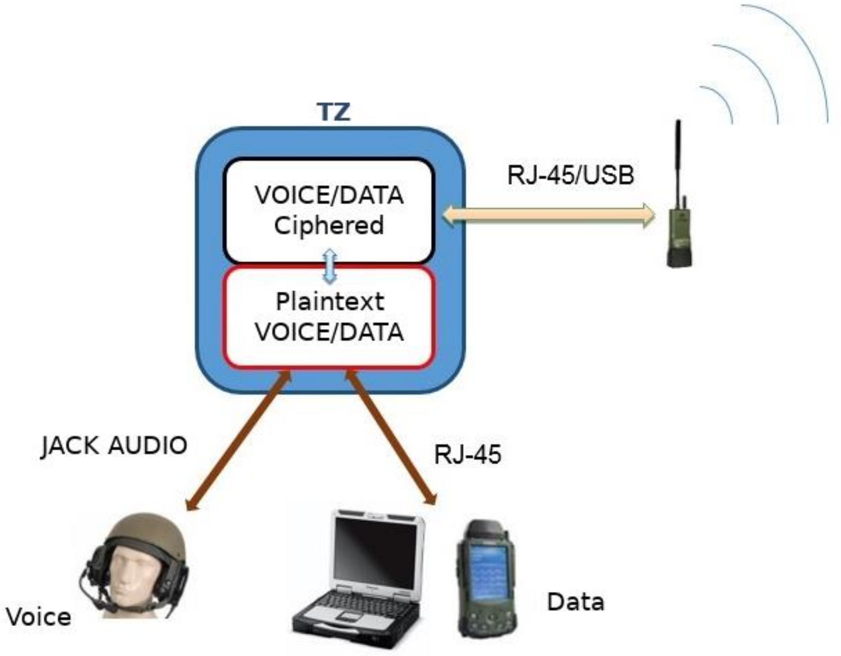


Figure 1 TZ use architecture

The operation system infrastructure is compound by the following elements:

- CMAP: Crypto management Application. It is the workstation that contains the software responsible for creating and managing all the cryptographic material deployed into the TZ devices, such as cryptographic keys and missions.
- Tactical IP radio: Device which transmit IP traffic to other stations over the air. Tested radios are Harris 7800, PR4G and Spearnet.
- Router: Instead using tactical IP radios it is possible to use routers to connect the different TZs.

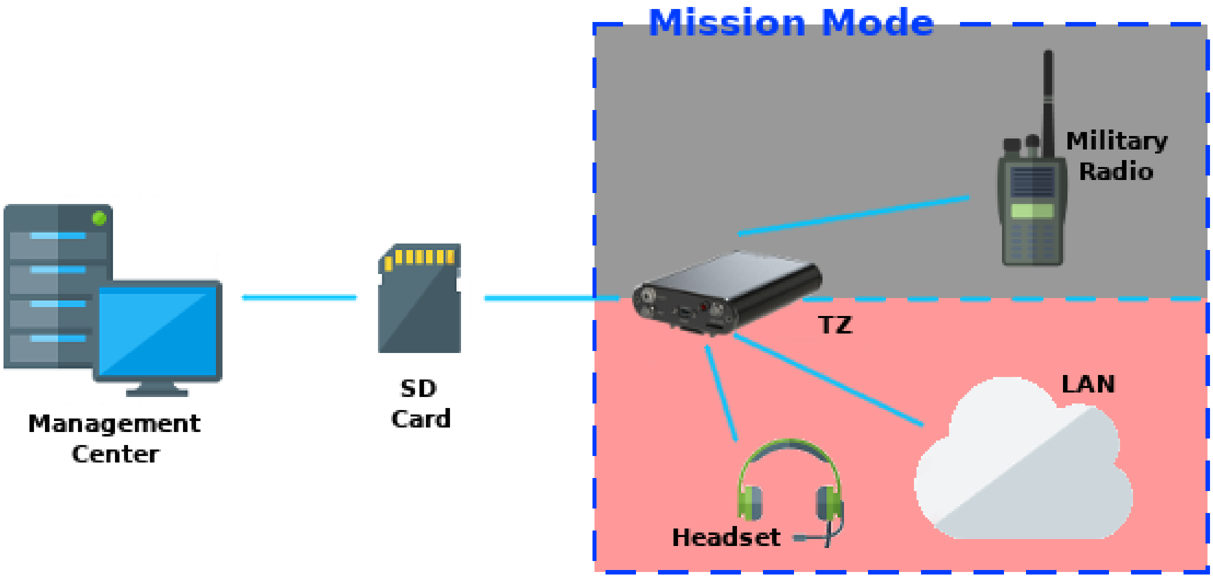


Figure 2 TZ architectural infrastructure

1.4 TOE Description

The TOE is defined as a set of hardware, software and firmware contained within a secured boundary. The boundary provides physical countermeasures to protect stored data from information disclosures or integrity breaches.

The TOE is logically defined by security functions. These functions are the following:

- Detection of logical breaches.
- Protection of stored data in TOE.
- Creation of security accounting to register security events.

Depending on the role using the TOE, it provides the usage of different features:

- Operator role: Encryption and decryption to protect the integrity and confidentiality of transmitted operator's data. Data is well-protected by using mission's cryptographic keys and mission data.

- Management role: this TOE configuration allows to management user to load new cryptographic keys, missions or software, by using the Management Centre.

1.4.1 Physical Scope

The physical boundary of the TOE is the TZ device.

1.4.1.1 Hardware

TZ is an isolated device with hardware, firmware and software to secure communications between different units.

Regarding the TOE internal hardware archictecture, it has two different zones inside: a red one and a black one. Each zone has its own board, being physically separated. Black zone manages the communication channel and the red zone is charged of encryption, decryption and processing of user data.

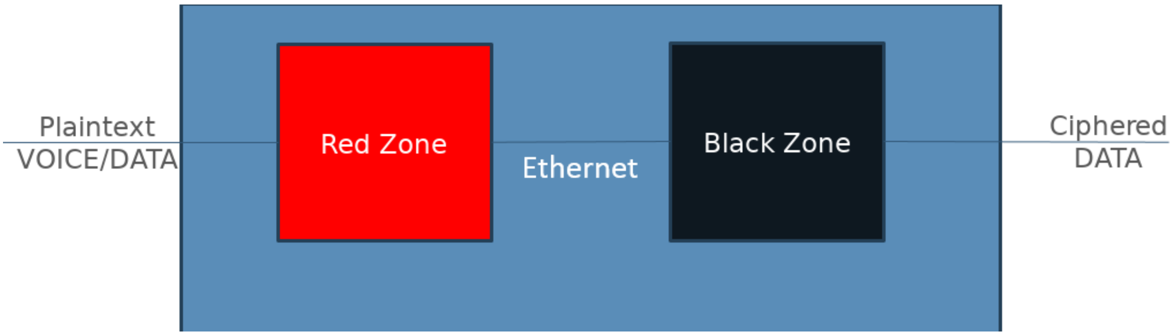


Figure 3 TZ red and back boards

The TOE boundary is depicted in Figure 4, where it is embeeded in a opaque box with security countermeasures.

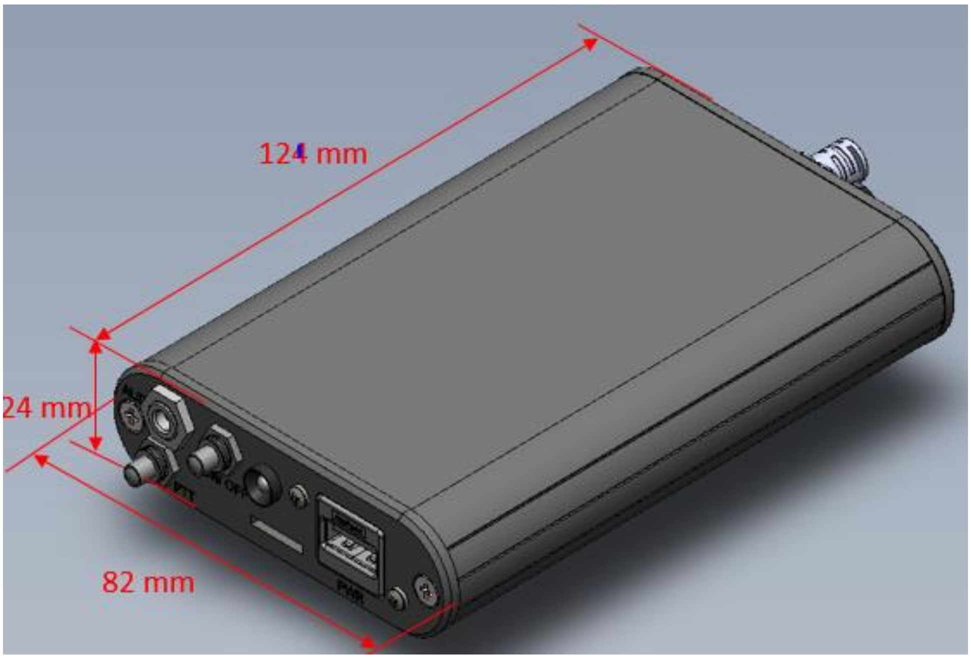


Figure 4 TZ device

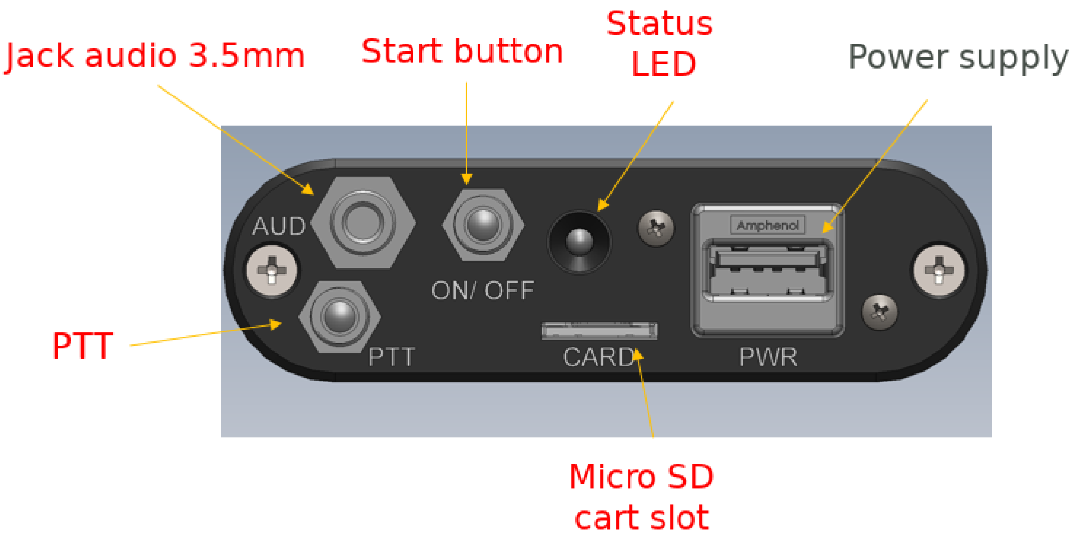


Figure 5 TZ front

Figure 5 shows the interfaces hosted in front of the TZ. These are the following:

- Start button powers on/off the system.
- LED status indicates the TZ current status even if TZ is zeroized.
- Micro-SD card slot allows to insert a micro-SD card containing mission data and mission's cryptography keys.
- Power supply slot supports the power input of the TZ device.

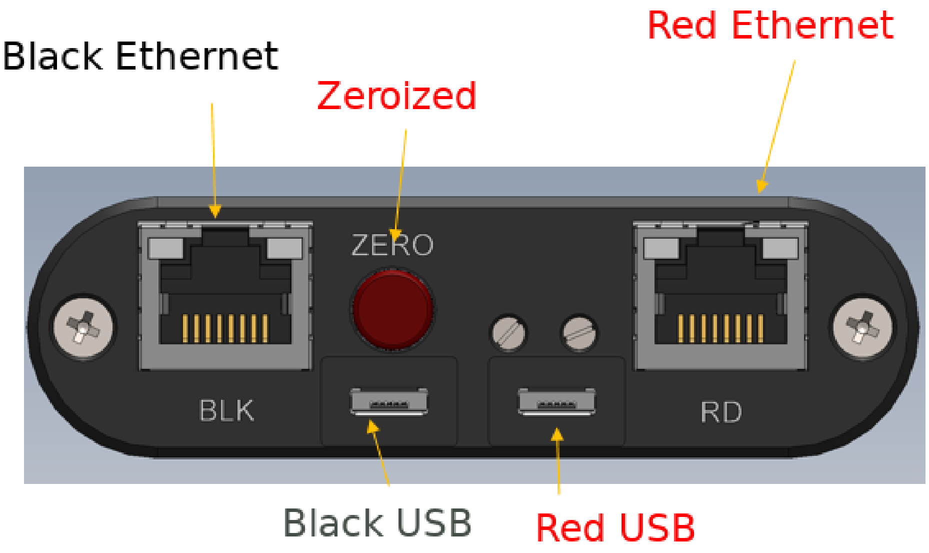


Figure 6 TZ back

Figure 6 depicts the TZ back face. This is composed by:

- A Black Ethernet slot that for sending/receiving cyphered data from Ethernet communication network.
- A red Ethernet slot, allowing user to send/receive non-cyphered data over Ethernet transport network. This port is also used for mainteinance purposes

by the maintenance role. For that, TOE should be connected to CMAP for carrying out maintenance task such as software updates and mission loadings.

- Zero button for zeroing the TZ device.

Note: Black USB and Red usb are not evaluated because they are out of the scope.

1.4.1.2 Software

The software architecture is the same for both boards (red and black). Both have a Linux operating system and they use C++ libraries.

The next figure describes the software architecture and the relation between both zones (red and black):

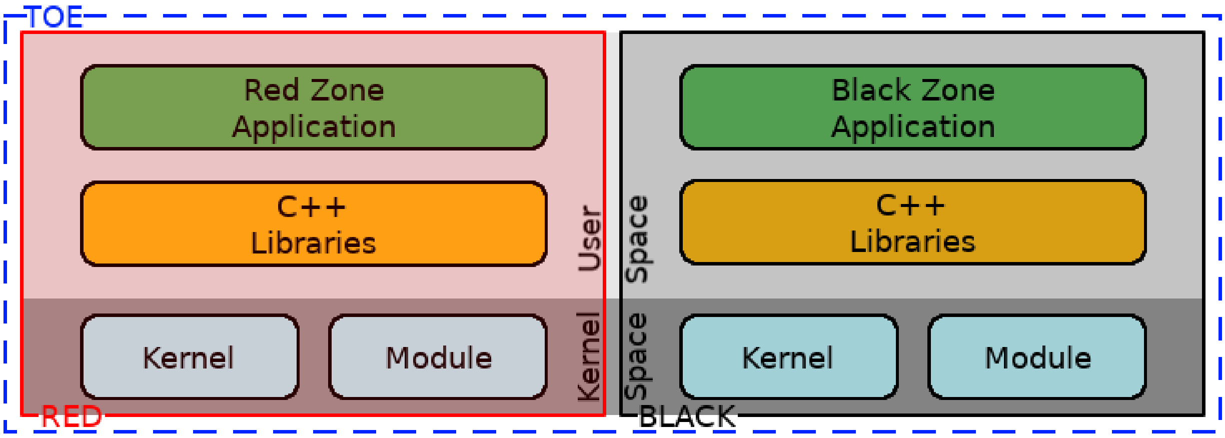


Figure 7 TZ Software architecture

1.4.2 TOE Delivery

TOE is delivered in an opaque package with a security seal.

If the customer detects any kind of modification, the package has to be returned to:

- Tecnobit. Cifpecom support.
- C/ Fudre, 18. 13.300 – Valdepeñas (Spain).

Adittionaly, a mail has to be sent to cifpecom@oesia.com indicating the problem with the package.

It is worth to noting that TOE version 2.4 is defined as:

- TOE software version 2.2
- Hardware version within P/N : 112EQ100001 Rev: 07,

so it is important to verify that they are included in TOE delivery. Firmware programmable logic version associated to hardware components is the same for P/N : 112EQ100001.

The package content is:

- TZ device (P/N : 112EQ100001 Rev: 07)
- Micro-SD card, precharged with a “dummy” mission version.
- TOE software version 2.2
- Documentation described in subsection 1.4.2.

When a new TOE software version is required, the new version will be sent to the customer in a CD/USB. An opaque package with a security seal will be used to protect the information.

If any kind of problem is detected, the customer has to report it to Tecnobit via mail: cifpecom@oesia.com

1.4.2.1 Documentation

The following list enumerates the guidance documents that constitute the TOE and that are provided to the consumer by encrypted email:

- T00740000ILS008 - Manual de Empleo y Mantenimiento
- T90421000PRE009 - Manual de configuración del entorno operacional
- T90421000DEL008 - Entrega del TOE
- T90421000FLR008 - Plan de Actualizaciones de Seguridad y solución de bugs de TZ.

The documentacion is delivered to the consumer in PDF format.

1.4.3 TOE Configuration

TOE is configured to work with the elements contained in delivery package as section 1.4.3 describes and within both operator and management roles. For interacting with TOE management capabilities, the Management Centre is required.

TOE is able to perform secure communications depending on the mission stored in the micro-SD. TOE is connected to a computer for data transmission and to a network device such as a switch, for sending data through the network.

1.4.4 Logical Scope

This section summarizes the security functions provided by this TOE:

1. Security audit
2. Cryptographic support
3. User data protection

4. Identification and authentication
5. Security management
6. Protection of the TSF
7. Trusted path/channels

1.4.4.1 Security audit

The TOE records and store events that happens in the system for reviewing. Furthermore, the system guarantees the availability, the integrity and the confidentiality of the logs.

1.4.4.2 Cryptographic support

The TOES uses cryptographic functions to encrypt and verify the information used by the system.

1.4.4.3 User data protection

The TOE has SFP to protect the security attributes, the access control and the information flow control.

1.4.4.4 Identification and authentication

The TOE uses IPSec capabilities to validate and authenticate the management centre. ESP (Encapsulated security payload) garantees integrity, confidenciality and authentication of the transmitted frames.

It also guarantees that only the adequate user can establish a communication with the management centre providing with mechanism to handle unauthorized attempts and avoiding any attempt to achive a MitM (Man in the middle) attack

1.4.4.5 Security management

The TOE implemented a user policy to restrict the access to the services, functions and attributes it provides. The policy is based on the following roles:

- Operator role: In this role, TOE is just able to transmit data based on the TOE cryptography configured in missions. This role is entered when TOE is powered-on whithout a management centre connected to.
- Management role: In this role, TOE is capable of updating cryptographic keys, missions and software. In addition, this role gets audit files from TOE. This role is reached when the TZ device is powered-on and the Management Centre is connected to the TOE.

1.4.4.6 Protection of the TSF

The TOE has mechanisms to protect the TSF against physical attacks and logical failures; and self-testing mechanism to check the system works properly.

1.4.4.7 Trusted path/channels

The TOE uses IPSec protocol to protect the communication channels between the TOE and other IT system.

2 CONFORMANCE CLAIMS

2.1 CC Conformance Claim

2.1.1 CC Version

This ST and TOE is compliant with Common Criteria for Information Technology Security Evaluation, version 3.1 revision 5. April 2017.

2.1.2 CC Part 2 Conformance Claims

This ST and TOE is Part 2 extended, being compliant with Common Criteria for Information Technology Security Evaluation, part 2 extended, version 3.1 revision 5. April 2017.

2.1.3 CC Part 3 Conformance Claims

This ST and TOE is Part 3 conformant, being compliant with Common Criteria for Information Technology Security Evaluation, part 3, version 3.1 revision 5. April 2017.

2.2 PP Claims

This ST does not claim conformance to any Protection Profile.

2.3 Package Claims

This ST claims conformance to EAL 2 augmented with ALC_FLR.1.

3 SECURITY PROBLEM DEFINITION

This section describes the security aspects of the operational environment and its expected use in said environment. It includes the declaration of the TOE operational environment that identifies and describes:

- The alleged known threats that will be countered by the TOE
- The organizational security policies that the TOE has to adhere to
- The TOE usage assumptions in the suggested operational environment.

We will begin defining Assets and Agents of threats.

3.1 Assets

AS.INFORMATION: The confidentiality, authenticity and integrity of data information travelling through the TOE between the red and black interfaces and through the black network.

AS.CONFIG: The confidentiality, authenticity and integrity of TOE configuration data, including cryptographic keys.

AS.LOGS: The confidentiality, authenticity and integrity of TOE audit logs.

AS.TOPOLOGY: The confidentiality of the network topology information.

AS.SOFTWARE: The confidentiality and integrity of TOE u-boot, SO and application software and their encryption algorithms.

3.2 Threat agents

This section depicts which agents can get involved in TOE security problems, relating them with defined threats to security. These users are defined according TOE' security functions that they can vulnerate.

S.ATTACKER_LOCAL: An attacker with physical access.

S.ATTACKER_RED: An attacker in the red/non-cyphered network , which can access it logically.

S.ATTACKER_BLACK: An attacker in the black/cyphered network, which can access it logically.

3.3 Threats

This section identifies the threats to assets that require protection by the TOE. The threats are defined in terms of assets concerned, attackers and the adverse action that materializes the threat.

T.LEAKAGE: ***S.ATTACKER_BLACK*** is able to modify/delete/disclosure the protected **AS.INFORMATION** or access the red network or is able to get information about **AS.TOPOLOGY**

T.MC_IMPERSONATE: ***S.ATTACKER_RED*** is able to impersonate the Management Centre gaining access to **AS.CONFIG** or subverting routing configuration.

T.PHYSICAL_TAMPER: An ***S.ATTACKER_LOCAL*** is able to tamper the TOE to get secrets, to modify data on whose integrity the TSF relies, or to corrupt or de-activate the TSF to violate, subvert or decrypt **AS.SOFTWARE** or **AS.CONFIG**

T.UNDETECTED_ACTIVITY: ***S.ATTACKER_LOCAL***, ***S.ATTACKER_RED*** or ***S.ATTACKER_BLACK*** may attempt to access, change, and/or modify the security functionality of the network device without management awareness.

This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the management role would have no knowledge that the device has been compromised (**AS.LOGS**).

T.MALFUNCTION: ***S.ATTACKER_LOCAL*** may use a malfunction of the hardware or software, which is accidental or deliberated by applying environmental stress or perturbation, in order to:

- Deactivate
- Modify
- Circumvent

security functions of the TOE to enable attacks against **AS.CONFIG**

T.MISS_CONFIGURATION: ***S.ATTACKER_LOCAL,*** ***S.ATTACKER_RED*** or ***S.ATTACKER_BLACK*** may take advantage of a miss configuration or an error in the configuration to gain access to **AS.CONFIG, AS.TOPOLOGY, AS.SOFTWARE.**

3.4 Organizational Security Policies

The organizational Security policies are defined as follows.

P.ROLES: TOE will support several roles. These are the following: **S.MANAGEMENT**, to configure and stablish keys and missions and to perform TOE software updates and to get and manage the TOE accounting; and **S.OPERATOR** role, which uses TOE' secure data communication capability.

P.LOGS: The users of the TOE shall be held accountable for their actions within the system.

3.5 Assumptions

The assumptions when using the TOE are the following:

A.MANAGEMENT: A Management Centre is required to access TZ management functions.

A.KEY_GENERATION: Cryptographic keys generated by the Management Centre and HSM and imported into the TOE are cryptographically strong enough for the intended key usage.

A.AUDIT_ANALYSIS: The Management Centre obtains audit records from the TOE and S.MANAGEMENT role analyses them for security violations.

A.USERS: TOE users are responsible to accomplish secure procedures according their responsabilities.

A.SECURE_ENVIRONMENT: It is assumed that functionalities affecting S.MANAGEMENT role will be achieved in a secure environment.

4 SECURITY OBJECTIVES

The security objectives are high level declarations, concise and abstract of the solution to the problem exposed in the former section, which counteracts the threats and fulfils the security policies and the assumptions. These consist of:

- The security objectives for the operational environment.
- The security objectives for the TOE.

4.1 Security Objectives for the TOE

The security objectives for the TOE must determine (to the desired extent) the responsibility of the TOE in countering the threats and in enforcing the OSPs.

Each objective must be traced back to aspects of identified threats to be countered by the TOE and to aspects of OSPs to be met by the TOE.

O.RED-BLACK: The TOE shall protect confidential information for export into the black area by encryption of plaintext data and for import into the red area by decryption of cipher text data.

The TOE shall protect integrity sensitive information for export into the black area and for import into the red area.

O.ROLES: The TOE shall provide the following roles:

- S.MANAGEMENT
- S.OPERATOR

Access to TOE services by the different roles shall be under an access control policy. Access to configuration by the different roles shall be under an access control policy.

O.AUDIT: The TOE shall provide the capability to detect and create audit records of security relevant events associated with users.

O.SDCONFIG: Configuration imported by the TOE through the micro-SD shall save the confidentiality and integrity.

O.DESTRUCTION: The TOE shall destruct, in a secure way, the cryptographic keys and other CSP on demand of authorized users or when they will not be used any more.

O.PHYSICAL: The TOE shall protect confidentiality of CSP under physical attacks.

O.SELFTEST: The TOE shall perform regular checks to verify that its components operate correctly. This includes integrity checks of TOE software, firmware, internal

TSF data and keys during initial start-up and at the conditions installation and maintenance.

O.MANAGEMENT: The TOE shall authenticate S.MANAGEMENT role using the Management Centre interface to manage the TOE configuration, import cryptographic keys and review audit logs.

O.FIRMWARE: Firmware upgrades shall be performed using the Management Centre interfaces with integrity, authenticity and confidentiality.

O.CRYPTO: The implemented cryptographic functions that support TOE operation shall be implemented, using known standards, with appropriate cryptographic strengh and shall relay in TOE trusted operating system.

4.2 Security Objectives for the Operational Environment

The security objectives for the Operational Environment determine the responsibility of the environment in countering the threats, enforcing the OSPs and upholding the assumptions.

Each objective must be traced back to aspects of identified threats to be countered by the environment, to aspects of OSPs to be enforced by the environment and to assumptions to be uphold by the environment.

OE.KEY_GENERATION: The IT environment shall ensure the cryptographic strength, the confidentiality and integrity of secret and private keys.

Furthermore, the integrity and authenticity of public keys generated outside the TOE and imported into the TOE.

OE.AUDIT_ANALYSIS: The TOE environment reviews the audit trails generated and exported from the TOE to detect security violation and making users accountable for their actions related to the TOE.

OE.PERSONAL: The S.OPERATOR and S.MANAGEMENT roles are assigned with distinct authorized persons. The personal is well trained and responsible in the fulfilment of their functions.

OE.MANAGEMENT: The Management Centre is used in a secure environment by trained personnel and provides well-formed configuration information to the TOE.

4.3 Security Objectives Rationale

The following table illustrates that the security objectives and the security objectives for the operational environment counter all threats, enforce all OSPs and uphold all assumptions.

	O.RED-BLACK	O.ROLES	O.AUDIT	O.SDCONFIG	O.DESTRUCTION	O.PHYSICAL	O.SELFTEST	O.MANAGEMENT	O.FIRMWARE	O.CRYPTO	OE.KEY_GENERATION	OE.AUDIT_ANALYSIS	OE.PERSONAL	OE.MANAGEMENT
T.LEAKAGE	X			X						X				
T.MC_IMPERSONATE								X		X				X
T.PHYSICAL_TAMPER					X	X			X					
T.UNDETECTED_ACTIVITY			X									X		
T.MALFUNCTION							X		X					
T.MISS_CONFIGURATION	X			X										
P.ROLES		X											X	
P.LOGS			X									X		
A.MANAGEMENT		X						X						X
A.KEY_GENERATION											X			
A.AUDIT_ANALYSIS												X		
A.USERS													X	
A.SECURE_ENVIRONMENT													X	X

Table 1 Security Objectives vs Security Problem Definition

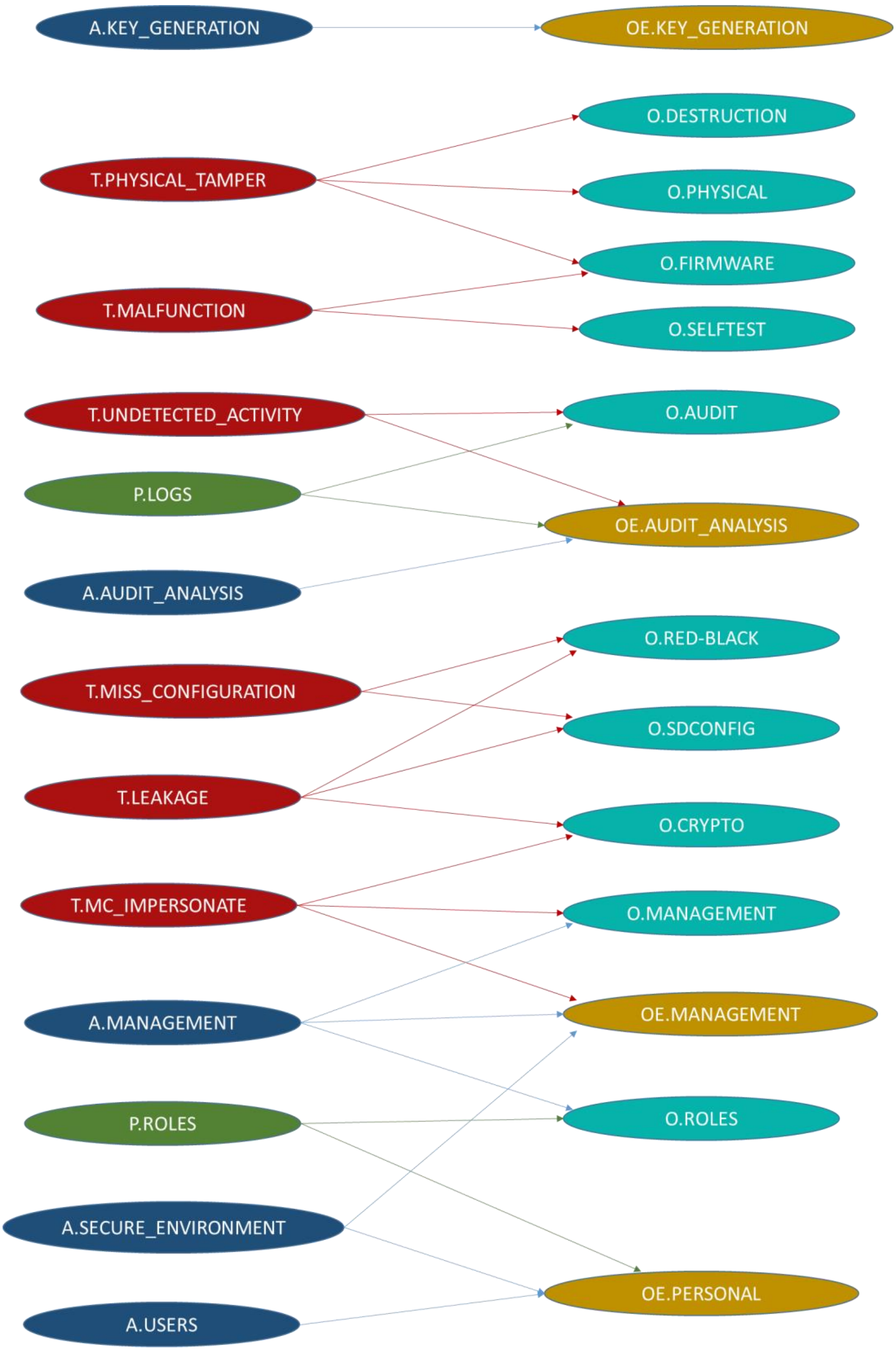


Figure 8 Mapping of Security Problem Definition to Security Objectives

4.3.1 Threats

T.LEAKAGE: Attacks coming from the black network are addressed by the TOE through the security objective ***O.RED-BLACK*** which ensures that unprotected data from the red network does not cross to the black network.

Ensuring that the configuration is protected also contributes to mitigate this threat (***O.SDCONFIG***).

O.CRYPTO contributes implementing strong cryptographic mechanisms.

T.MC_IMPERSONATE: This threat is addressed by ***O.MANAGEMENT*** that requires the TOE to authenticate the access to the management interface and by **OE.MANAGEMENT** that ensures a correct and secure use of the management centre.

O.CRYPTO contributes implementing strong cryptographic mechanisms.

T.PHYSICAL_TAMPER: This threat describes tampering the cryptographic module

- To get secrets
- To modify data on whose integrity the TSF relies
- To corrupt or de-activate the TSF inside the cryptographic boundary

The security objective ***O.PHYSICAL*** and ***O.DESTRUCTION*** address this threat.

O.FIRMWARE contribute ensuring that firmware upgrades loaded through the maintenance interface are integrity and authentic.

T.UNDETECTED_ACTIVITY: The TOE security is required to provide the capability to detect and create audit records of security relevant events associated with users. The objective ***O.AUDIT*** address this threat.

The security objective for the IT environment ***OE.AUDIT_ANALYSIS*** ensures reviews of the audit trails generated and exported from the TOE, ensuring this threat is mitigated.

T.MALFUNCTION: This threat describes the use of a malfunction of the hardware or software in order to:

- Deactivate
- Modify
- circumvent security functions of the TOE to enable attacks against the integrity or confidentiality of the User data or the CSP.

The security objective ***O.SELFTEST*** prevents this threat by regular checks verifying that TOE components operate correctly.

O.FIRMWARE contribute ensuring that firmware upgrades loaded through the maintenance interface are integrity and authentic.

T.MISS_CONFIGURATION: This threat involves the configuration of some required files in order to perform communication among red and black parts, and the correctness in files stored in the SD.

Thus, the security objective ***O.SDCONFIG*** will prevent the bad configuration in any contained file; and the security objective ***O.RED-BLACK*** will support and control that Black and Red elemements are well-configured.

The following table maps the threats of the security problem established to the security objectives of the TOE and the security objectives of the operational environment.

Threats	Security Objectives
T.LEAKAGE	O.RED-BLACK O.SDCONFIG O.CRYPTO
T.MC_IMPERSONATE	O.MANAGEMENT O.CRYPTO OE.MANAGEMENT
T.PHYSICAL_TAMPER	O.PHYSICAL O.DESTRUCTION O.FIRMWARE
T.UNDETECTED_ACTIVITY	O.AUDIT OE.AUDIT_ANALYSIS
T.MALFUNCTION	O.SELFTEST O.FIRMWARE
T.MISS_CONFIGURATION	O.RED-BLACK O.SDCONFIG

Table 2 Threats vs Security Objectives

4.3.2 Organisational Security Policies

P.ROLES: This organisational security policy addresses separate and distinct roles for Management (S.MANAGEMENT) and for operators (S.OPERATOR).

The security objective ***O.ROLES*** requires the TOE to implement them and the security objective **OE.PERSONAL** requires the IT environment to use them.

P.LOGS: This organisational security policy requires the users be held accountable for their actions within the system.

The TOE security is required to provide the capability to detect and create audit records of security relevant events associated with users by the objective ***O.AUDIT***.

The security objective for the IT environment **OE.AUDIT_ANALYSIS** ensures reviews of the audit trails generated and exported from the TOE, making authenticated users accountable for their actions related to the TOE.

The following table maps the organizational security policies of the problem established to the security objectives of the TOE and the security objectives of the operational environment.

OSPs	Security Objectives
P.ROLES	O.ROLES OE.PERSONAL

OSPs	Security Objectives
P.LOGS	O.AUDIT OE.AUDIT_ANALYSIS

Table 3 OSPs vs Security Objectives

4.3.3 Assumptions

A.MANAGEMENT: This assumption is fully covered by the security objective for the IT environment **OE.MANAGEMENT,** ***O.MANAGEMENT*** and ***O.ROLES*****.**

A.KEY_GENERATION: This assumption deals with the cryptographic strength and security attributes of cryptographic keys generated by Management Centre and imported into the TOE.

This assumption is directly and completely covered by the security objective for the IT environment **OE.KEY_GENERATION.**

A.AUDIT_ANALYSIS: This assumption addresses reading and analysis of audit records of the TOE as implemented by **OE.AUDIT_ANALYSIS**

A.USERS: This assumption is fully covered by the security objective for the IT environment **OE.PERSONAL**, which deals with the training of the users and the separation of privileges.

A.SECURE_ENVIRONMENT. This assumption is fully covered by the security objectives: **OE.MANAGEMENT** garantees that the management centre is used in a secure environment. Also, the **OE.PERSONAL** implies that the management centre capabilities are only used by the security personal.

The following table maps the assumptions of the problem established to the security objectives of the TOE and the security objectives of the operational environment.

Assumptions	Security Objectives
A.MANAGEMENT	OE.MANAGEMENT O.MANAGEMENT O.ROLES
A.KEY_GENERATION	OE.KEY_GENERATION
A.AUDIT_ANALYSIS	OE.AUDIT_ANALYSIS
A.USERS	OE.PERSONAL
A.SECURE_ENVIRONMENT	OE.MANAGEMENT OE.PERSONAL

Table 4 Assumptions vs Security Objectives for the Operational Environment

5 EXTENDED COMPONENTS DEFINITION

5.1 Extended Security Functional Requirements

The Security Functional Requirements included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, with additional extended functional components:

- FPT_TST.2 TSF Self-Testing
- FPT_PHP.4 Opaque Encapsulation
- FPT_TOS.1 Trusted Operating System
- FIA_509.1 X.509 Certificate Validation
- FIA_509.2 X.509 Certificate Authenticaton

5.1.1 Class FPT: Protection of the TSF

This class contains families of functional requirements that relate to the integrity and management of the mechanisms that constitute the TSF and to the integrity of TSF data.

In some sense, families in this class may appear to duplicate components in the FDP class. They may even be implemented using the same mechanisms. However, FDP focuses on user data protection, while FPT focuses on TSF data protection. In fact, components from the FPT class are necessary to provide requirements that the SFPs in the TOE cannot be tampered with or bypassed.

From the point of view of this class, regarding to the TSF there are three significant elements:

- The TSF's implementation, which executes and implements the mechanisms that enforce the SFRs.
- The TSF's data, which are the administrative databases that guide the enforcement of the SFRs.
- The external entities that the TSF may interact with in order to enforce the SFRs.

5.1.1.1 TSF self test (FPT_TST)

Family behaviour

The family defines the requirements for the self-testing of the TSF with respect to some expected correct operation.

Examples are interfaces to enforcement functions, and sample arithmetical operations on critical parts of the TOE.

These tests can be carried out at:

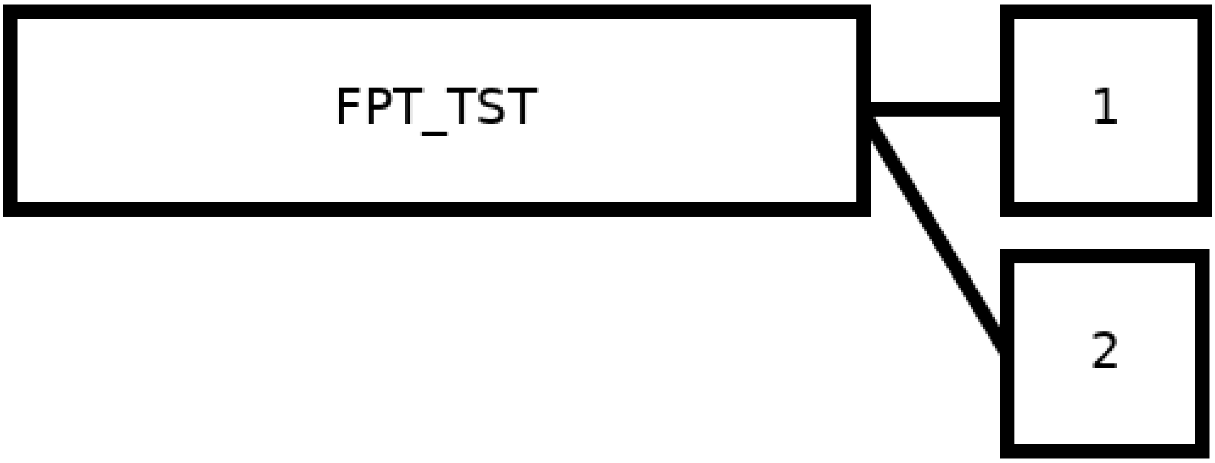
- start-up
- periodically
- at the request of the authorised user
- when other conditions are met.

The actions to be taken by the TOE as the result of self testing are defined in other families.

The requirements of this family are also needed to detect the corruption of TSF data and TSF itself (i.e. TSF executable code or TSF hardware component) by various failures that do not necessarily stop the TOE's operation (which would be handled by other families).

These checks must be performed because these failures may not necessarily be prevented. Such failures can occur either because of unforeseen failure modes or associated oversights in the design of hardware, firmware, or software, or because of malicious corruption of the TSF due to inadequate logical and/or physical protection.

Component levelling



FPT_TST.2 TSF self-testing requires self-testing capabilities of the TSF correct operation. These tests must be performed at start-up. Conditional and on demand by a user self-testing may be required. Particular TSF behaviour during self-testing and TSF-actions after self-testing are required.

FPT_TST.2 inherits from FPT_TST.1, adding derived actions from the execution of self-testing capabilities.

Management: FPT_TST.2

There are no management activities foreseen.

Audit: FPT_TST.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the ST:

a) Basic: Execution of the TSF self tests and the results of the tests.

FPT_TST.2 TSF self-testing

Hierarchical to: No other components

Dependencies: FPT_FLS.1 Failure with preservation of secure state.

FPT_TST.2.1 The TSF shall perform self-testing at power-up to verify the correctness of *[assignment: list of cryptographic algorithms*] and of [*assignment: list of critical TSF*], and to verify the integrity of the TSF-software/firmware.

FPT_TST.2.2 The TSF shall perform self-testing at the conditions [*assignment: list of conditions*] to verify the correctness of [*assignment: list of critical cryptographic algorithms*].

FPT_TST.2.3 The TSF shall perform self-testing at the conditions [*assignment: list of conditions*] to verify the correctness of [*assignment: list of critical TSF*], and to verify the integrity of [*assignment: list of TSF data*].

FPT_TST.2.4 The TSF shall perform self-testing at the conditions [*assignment: list of conditions*] to verify the integrity of [*assignment: list of TSF-objects*].

FPT_TST.2.5 The TSF shall provide [*assignment: list of users*] with the capability to invoke the following self-tests [*assignment: list of self-tests*].

FPT_TST.2.6 After completion of self-testing the TSF shall [*assignment: list of actions to be performed*].

FPT_TST.2.7 If the self-testing result is fail the TSF shall [*assignment: list of actions to be performed*].

5.1.1.2 TSF Physical Protection (FPT_PHP)

Family behaviour

TSF physical protection components refer to restrictions on unauthorised physical access to the TSF. It also refers to the deterrence and resistance to unauthorised physical modification or substitution of the TSF.

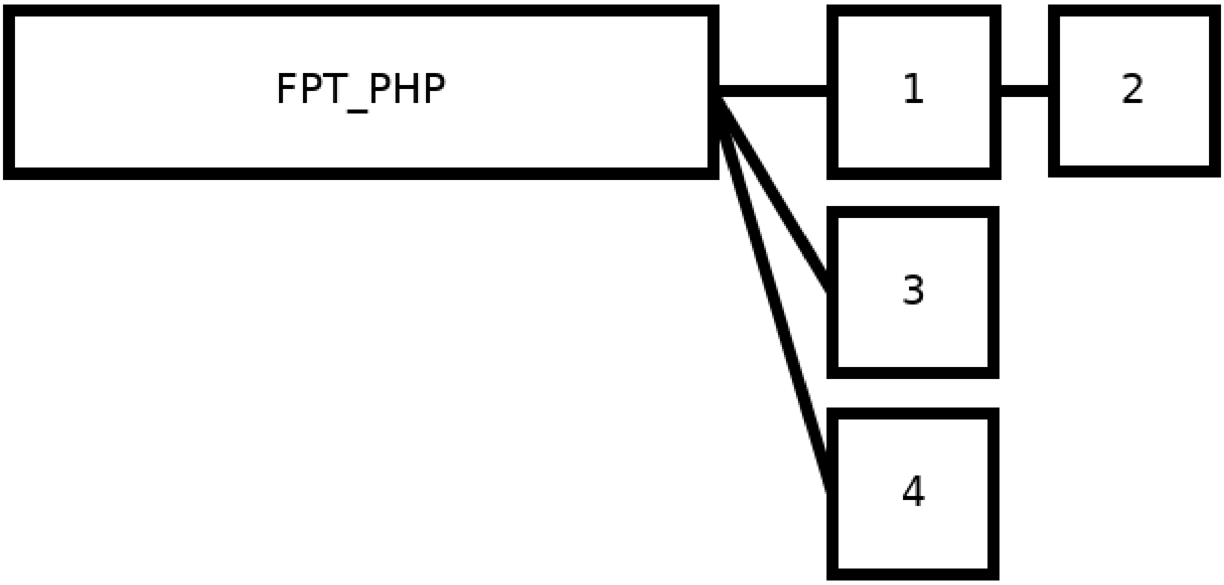
The requirements of components in this family ensure that the TSF is protected from physical tampering and interference.

Satisfying the requirements of these components results in the TSF being packaged and used in such a manner that physical tampering is detectable, or resistance to physical tampering is enforced.

Without these components, the protection functions of a TSF lose their effectiveness in environments where physical damage cannot be prevented.

This family also provides requirements regarding how the TSF shall respond to physical tampering attempts.

Component levelling



FPT_PHP.4 extends FPT_PHP family, specifying a construction requirement that forces the box enclosing the TOE to be opaque and physically continuous.

Management: FPT_PHP.4

There are no management activities foreseen.

Audit: FPT_PHP.4

There are no auditable events foreseen.

FPT_PHP.4 Opaque Encapsulation

Hierarchical to: No other components

Dependencies: No dependencies

FPT_PHP.4.1	The TOE enclosure shall be opaque and physically continuous, so the direct observation or manipulation of inside the module is not possible.

5.1.1.3 Trusted Operating System (FPT_TOS)

Family behaviour

This family defines the requirements to consider trusted the underlying operating system of the TOE, avoiding unauthorised physical modification or substitution of that component

Satisfying the requirements of these components results in the TSF being packaged, stored in TOE and secure booted on TOE start-up such a manner that a modification, substitution or deletion of the operating system is detectable.

This family also provides requirements regarding cryptographic operations performed during the integrity, authentication and booting process.

Component levelling

FPT_TOS.1 focuses on the TOE operating system must be integral and securely stored. Strong authentication mechanisms shall be used. Proccess shall guarantee that operating system loaded in TOE is loaded though a secure boot process from a controlled source by verifying integrity and authentication during the secured booting process.

Management: FPT_TOS.1

There are no management activities foreseen.

Audit: FPT_TOS.1

There are no auditable events foreseen.

FPT_TOS.1 Trusted Operating System

Hierarchical to: No other components

Dependencies: No dependencies

FPT_TOS.1.1	The underlying operating system must boot from a controlled source.
FPT_TOS.1.2	The underlying operating system integrity must be verified during boot.
FPT_TOS.1.3	The underlying operating system must have strong authentication mechanisms.

5.1.2 Class FIA: Identification and authentication

Families in this class address the requirements for functions to establish and verify a claimed user identity.

Identification and Authentication is required to ensure that users are associated with the proper security attributes (e.g. identity, groups, roles, security or integrity levels).

The unambiguous identification of authorised users and the correct association of security attributes with users and subjects is critical to the enforcement of the intended security policies.

The families in this class deal with determining and verifying the identity of users, determining their authority to interact with the TOE, and with the correct association of security attributes for each authorised user.

Other classes of requirements (e.g. User Data Protection, Security Audit) are dependent upon correct identification and authentication of users in order to be effective.

5.1.2.1 Authentication using X.509 certificates (FIA_509)

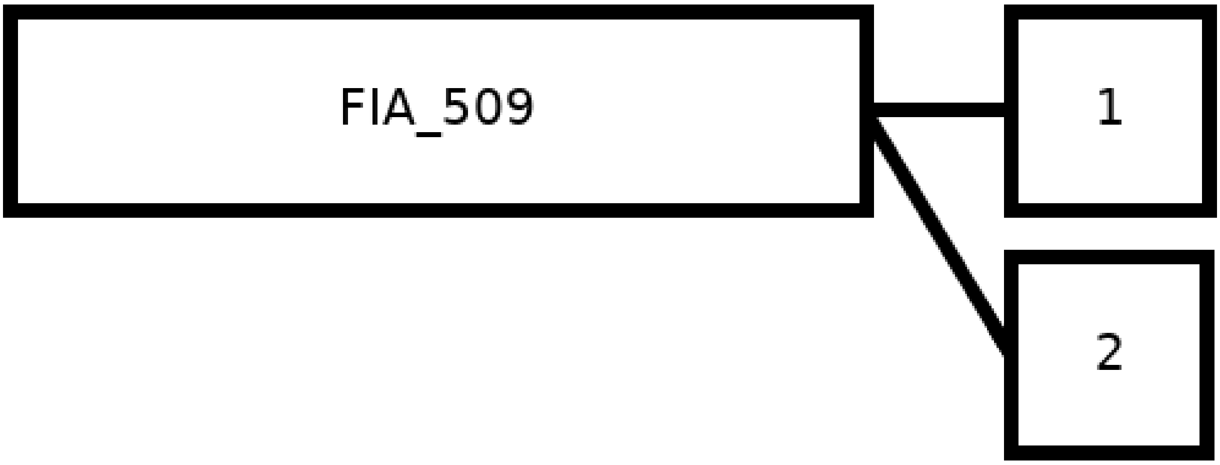
Family behaviour

This family defines the behaviour, management, and use of X.509 certificates for functions to be performed by the TSF.

Components in this family require validation of certificates according to a specified set of rules and use of certificates for authentication for protocols and integrity verification.

Satisfying the requirements of these components, several certificates are involved in micro-SD, Management Centre commands and application validation.

Component levelling



FIA_509.1 X509 Certificate Validation, requires the TSF to check and validate certificates in accordance with the RFCs and rules specified in the component.

FIA_509.2 X509 Certificate Authentication requires the TSF to use certificates to authenticate peers in protocols that support certificates, as well as for integrity verification and potentially other functions that require certificates.

Management: FIA_509.1

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

a) Remove imported X.509v3 certificates.

b) Approve import and removal of X.509v3 certificates.

Management: FIA_509.2

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

a) Remove imported X.509v3 certificates.

b) Approve import and removal of X.509v3 certificates.

Audit: FIA_509.1, FIA_509.2

There are no auditable events foreseen.

FIA_509.1 X.509 Certificate Validation

Hierarchical to: No other components

Dependencies: No dependencies

FPT_509.1.1	The TSF shall validate certificates in accordance with the following rules: – RFC 5280 certificate validation and certificate path validation.

- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basic Constraints extension and that the CA flag is set to TRUE for all CA certificates.

FPT_509.1.2 The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_509.2 X.509 Certificate Authentication

Hierarchical to: No other components

Dependencies: No dependencies

FPT_509.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [selection: *IPsec, TLS, [assignment: other protocols], no protocols]* and [selection: *code signing for system software updates, code signing for integrity verification, [assignment: other uses], no additional uses]*

FPT_509.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [selection: *allow the S.MANAGEMENT role to choose whether to accept the certificate in these cases, accept the certificate, not accept the certificate*].

5.2 Extended Security Assurance Requirements

There is not Extended Security Assurance Requirements included in this ST.

6 SECURITY REQUIREMENTS

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The notation, formatting, and conventions used in this Security Target are consistent with those used in Version 3.1 of the Common Criteria.

Selected presentation choices are discussed here to aid the Security Target reader.

The Common Criteria allows several operations to be performed on functional requirements. The allowable operations defined in Part 2 of the Common Criteria are:

- **Refinement operation** (denoted by bold text): is used to add details to a requirement, and thus further restricts a requirement.
- ***Selection*** (denoted by italicized text): is used to select one or more options provided by the [CC31R5P2] in stating a requirement.
- **Assignment operation** (denoted by italicized text between brackets): is used to assign a specific value to an unspecified parameter, such as the length of a password. Showing the value in square brackets indicates assignment.
- **Iteration operation**: are identified with the symbol "/" and an "identifier" following requirements identifier (e.g. "/XXX"). The iteration operation may be sometimes used just for the sake of clarity.

6.1 Security Functional Requirements

The following table list the SFRs claimed by the TOE.

Functional Class	**Functional Components**
FAU: Security audit	FAU_GEN.1 Audit data generation
	FAU_SAR.1 Audit review
	FAU_STG.2 Guarantees of audit data availability
	FAU_STG.4 Prevention of audit data loss
FCS: Cryptographic support	FCS_CKM.2 Cryptographic key distribution
	FCS_CKM.4 Cryptographic key destruction

	FCS_COP.1 Cryptographic operation
FDP: User data protection	FDP_ACC.2/SERVICES Complete access control
	FDP_ACC.2/CONFIG Complete access control
	FDP_ACF.1/SERVICES Security attribute based access control
	FDP_ACF.1/CONFIG Security attribute based access control
	FDP_IFC.2 Complete information flow control
	FDP_IFF.1 Simple security attributes
	FDP_ITC.2 Import of user data with security attributes
	FDP_UCT.1 Basic data exchange confidentiality
	FDP_UIT.1 Data exchange integrity
FIA: Identification and authentication	FIA_AFL.1 Authentication failure handling
	FIA_UAU.1 Timing of authentication
	FIA_509.1 X.509 Certificate validation
	FIA_509.2 X.509 Certificate authentication
FMT: Security management	FMT_MOF.1/AdminAct Management of security functions behaviour
	FMT_MOF.1/TrustedUpdate Management of security functions behaviour
	FMT_MSA.1/SERVICES Management of security attributes
	FMT_MSA.1/CONFIG Management of security attributes
	FMT_MSA.1/RED-BLACK Management of security attributes
	FMT_MSA.3/SERVICES Static attribute initialisation

	FMT_MSA.3/CONFIG Static attribute initialisation
	FMT_MSA.3/RED-BLACK Static attribute initialisation
	FMT_MTD.1 Management of TSF data
	FMT_SMF.1 Specification of management functions
	FMT_SMR.1 Security roles
FPT: Protection of the TSF	FPT_FLS.1 Failure with preservation of secure state
	FPT_PHP.1 Passive detection of physical attack
	FPT_PHP.3 Resistance to physical attack
	FPT_PHP.4 Opaque encapsulation
	FPT_STM.1 Reliable time stamps
	FPT_TDC.1 Inter-TSF basic data TSF data consistency
	FPT_TST.2 TSF self-testing
	FPT_TOS.1 Trusted operating system
FTP: Trusted path/channels	FTP_ITC.1 Inter-TSF trusted channel
	FTP_TRP.1 Trusted path

6.1.1 Security audit

6.1.1.1 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 *not specified* level of audit; and

c) [

- *Self-test execution*
- *Activation of anti-tamper mechanisms*
- *Destruction of cryptographic keys (FCS_CKM.4)*
- *Loading of configuration file*

- *Role change*
- *Authentication failure handling (FIA_AFL.1): the reaching of the threshold for the unsuccessful authentication attempts*

].

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

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, *[none]*.

6.1.1.2 FAU_SAR.1 Audit review

FAU_SAR.1.1 The TSF shall provide *[S.MANAGEMENT]* with the capability to read *[all the information]* from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

6.1.1.3 FAU_STG.2 Guarantees of audit data availability

FAU_STG.2.1 The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

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

FAU_STG.2.3 The TSF shall ensure that *[75% of memory used for]* stored audit records will be maintained when the following conditions occur: *audit storage exhaustion*.

6.1.1.4 FAU_STG.4 Prevention of audit data loss

FAU_STG.4.1 The TSF shall "*prevent audited events, except those taken by the authorised user with special rights*" and *[stop S.OPERATOR functionality]* if the audit trail is full.

6.1.2 Cryptographic support

6.1.2.1 FCS_CKM.2 Cryptographic key distribution

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method *[OTAR (over the air rekey)]* that meets the following:

[1) IPSec layer

2) SOG-is agreed cryptographic mechanism 1.0

3) Cryptographic key sizes [256].

]

Application Note: This SFR models the re-key functionality for data keys.

6.1.2.2 FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *[one or more, direct overwrite consisting of zeroes, followed by a read-verify. If the read-verification of the overwritten data fails, the process shall be repeated again until a number of tries greater than zero, where an error will be return]* that meets the following: *[No standards]*.

6.1.2.3 FCS_COP.1 Cryptographic operation

FCS_COP.1.1 The TSF shall perform *[the cryptographic operations described in the application note]* in accordance with a specified cryptographic algorithm *[the cryptographic algorithms described in the application note]* and cryptographic key sizes *[the cryptographic key sizes described in the application note]* that meet the following: *[the standards described in the application note]*.

Application Note:

Operation	Algorithm	Key size	Standard	Purpose
encryption/decryption	IPSEC (AES-GCM) (AES-CBC)	256	RFC4106 / RFC3602	Data information transfer

encryption/decryption	AES - ECB	256	NIST-Announcing the Advanced Encryption Standard (AES) (www.nvlpubs.nist.gov/nistpubs/fips/nist.fips.197.pdf)	Rekey distribution
Encryption/decryption	AES-GCM	256	NIST – The Galoise/Counter Mode of Operation (www.mindspring.com/~dmcgrew/gcm-nist-6.pdf)	Mission download package
encryption/decryption	AES - CCM	256	NIST Advanced Encryption Standard –AES-(www.nvlpubs.nist.gov/nistpubs/fips/nist.fips.197.pdf)	OS and software application
Digest	ECDSA	Brainpool-384	RFC6979 / RFC5639	TOE SW (TZ Red binary)
encryption/decryption	AES - GCM	256	NIST – The Galoise/Counter Mode of Operation (www.mindspring.com/~dmcgrew/gcm-nist-6.pdf)	Logs
encryption/decryption	AES - GCM	256	NIST – The Galoise/Counter Mode of Operation (www.mindspring.com/~dmcgrew/gcm-nist-6.pdf)	Configuration
encryption/decryption	AES - GCM	256	NIST – The Galoise/Counter Mode of Operation (www.mindspring.com/~dmcgrew/gcm-nist-6.pdf)	Keys
encryption/decryption	AES - GCM	256	NIST – The Galoise/Counter Mode of Operation (www.mindspring.com/~dmcgrew/gcm-nist-6.pdf)	Certificates
Encryption/decyption	AES-GCM	256	NIST – The Galoise/Counter Mode of Operation (www.mindspring.com/~dmcgrew/gcm-nist-6.pdf)	Software update package

6.1.3 User data protection

6.1.3.1 FDP_ACC.2/SERVICES Complete access control

FDP_ACC.2.1/SERVICES The TSF shall enforce the *[TOE Services access control SFP]* on *[*

a) Objects: TOE services,
- *Show Status*
- *Key import*
- *Configuration (FMT_SMF.1)*
- *Encrypt/Decrypt data*
- *Audit review*
- *System Maintenance*

b) Subjects: TOE users,
- *S.OPERATOR,*
- *S.MANAGEMENT*

]

and all operations among subjects and objects covered by the SFP.

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

6.1.3.2 FDP_ACC.2/CONFIG Complete access control

FDP_ACC.2.1/CONFIG The TSF shall enforce the *[Configuration access control SFP]* on *[*

a) Subjects: TOE Users,
- *S.MANAGEMENT*

b) *Objects: SD Configuration*

]

and all operations among subjects and objects covered by the SFP.

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

6.1.3.3 FDP_ACF.1/SERVICES Security attribute based access control

FDP_ACF.1.1/SERVICES The TSF shall enforce the *[TOE Services access control SFP]* to objects based on the following: *[*

a) Objects: TOE services with the following security attributes: service name and access control rights

b) Subjects: users with the following security attributes:

- *S.MANAGEMENT*
- *S.OPERATOR*

].

FDP_ACF.1.2/SERVICES The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *[Application note table]*.

Application Note:

Service	Roles
Show status	Any
Key loading	S.MANAGEMENT
Configuration	S.MANAGEMENT
Encrypt/Decrypt data	S.OPERATOR
Audit review	S.MANAGEMENT
System Maintenance & FW/SW upgrade	S.MANAGEMENT

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

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

6.1.3.4 FDP_ACF.1/CONFIG Security attribute based access control

FDP_ACF.1.1/CONFIG The TSF shall enforce the *[Configuration Access control SFP]* to objects based on the following: *[*

a) Subjects: Users with security attribute: S.MANAGEMENT.

b) *Objects: SD Configuration file without security attributes*

].

FDP_ACF.1.2/CONFIG The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *[none]*.

FDP_ACF.1.3/CONFIG The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *[User role is S.MANAGEMENT]*.

FDP_ACF.1.4/CONFIG The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *[User role is not S.MANAGEMENT]*.

6.1.3.5 FDP_IFC.2 Complete information flow control

FDP_IFC.2.1 The TSF shall enforce the *[Red-black separation SFP]* on *[the users subjects and DATA information]* and all operations that cause that information to flow to and from subjects covered by the SFP.

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

6.1.3.6 FDP_IFF.1 Simple security attributes

FDP_IFF.1.1 The TSF shall enforce the *[Red-black separation SFP]* based on the following types of subject and information security attributes: *[*

a) Subjects: users with security attribute role: S.OPERATOR,

b) Information: DATA with security attributes destination and encryption key

].

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

a) Send operation (transfer from red to black):

- *user role must be S.OPERATOR and DATA destination must be described in the Configuration.*
- *DATA information must be encrypted with the keys specified in the Configuration before crossing to the black interface*

b) Receive operation (transfer from black to red):

- *user role must be S.OPERATOR and DATA destination must be described in the Configuration.*
- *DATA information must be decrypted with the keys specified in the Configuration after crossing to the red interface*

].

FDP_IFF.1.3 The TSF shall enforce the *[none]*.

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

FDP_IFF.1.5 The TSF shall explicitly deny an information flow based on the following rules: *[output of plaintext information through the black interfaces]*.

6.1.3.7 FDP_ITC.2 Import of user data with security attributes

FDP_ITC.2.1 The TSF shall enforce the *[Configuration Access Control SFP]* when importing user data, controlled under the SFP, from outside of the TOE.

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

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

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

FDP_ITC.2.5 The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *[all keys imported controlled by the TSF shall be encrypted or entered using split knowledge procedures]*.

Application Note: This SFR models the keys import process

6.1.3.8 FDP_UCT.1 Basic data exchange confidentiality

FDP_UCT.1.1 The TSF shall enforce the *[Red-Black separation SFP]* by providing the ability to *transmit, receive* user data in a manner protected from unauthorised disclosure.

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

6.1.3.9 FDP_UIT.1 Data exchange integrity

FDP_UIT.1.1 The TSF shall enforce the [*Red-black separation SFP]* to [*transmit, receive]* user data in a manner protected from [*modification, deletion, insertion, replay]* errors.

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

6.1.4 Identification and authentication

6.1.4.1 FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1 The TSF shall detect when *[3]* unsuccessful authentication attempts occur related to *[micro-SD certificates and data authentication]*.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall *[zeroize CSPs]*.

6.1.4.2 FIA_UAU.1 Timing of authentication

FIA_UAU.1.1 The TSF shall allow *[the functionality available for implicitly assumed roles according to FMT_SMR.1]* on behalf of the user to be performed before the user is authenticated.

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

6.1.4.3 FIA_509.1 X.509 Certificate validation

FIA_509.1.1 The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.

- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.

FIA_509.1.2 The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

6.1.4.4 FIA_509.2 X.509 Certificate authentication

FIA_509.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [*no protocols*] and for the SD validation and the command integrity verification from CMAP.

FIA_509.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [*accept the certificate]*.

Application Note: This SFR models authentication of the SD and commands from Management Centre against the TOE.

6.1.5 Security management

6.1.5.1 FMT_MOF.1/AdminAct Management of security functions behaviour

FMT_MOF.1.1/AdminAct The TSF shall restrict the ability to *[modify the behaviour of]* the functions *[TOE Security Functions]* to *[S.MANAGEMENT]*.

Application Note: This functionality can be just accesed by using the Management Centre through the management interface.

6.1.5.2 FMT_MOF.1/TrustedUpdate Management of security functions behaviour

FMT_MOF.1.1/TrustedUpdate The TSF shall restrict the ability to *[enable]* the functions *[FW/SW update]* to *[S.MANAGEMENT]*.

6.1.5.3 FMT_MSA.1/SERVICES Management of security attributes

FMT_MSA.1.1/SERVICES The TSF shall enforce the *[TOE Services access control SFP]* to restrict the ability to *modify* the security attributes *[none]* to *[no one]*.

6.1.5.4 FMT_ MSA.1/CONFIG Management of security attributes

FMT_MSA.1.1/CONFIG The TSF shall enforce the *[Configuration Access Control SFP]* to restrict the ability to *modify* the security attributes *[none]* to *[no one]*.

6.1.5.5 FMT_ MSA.1/RED-BLACK Management of security attributes

FMT_MSA.1.1/RED-BLACK The TSF shall enforce the *[Red-black separation SFP]* to restrict the ability to *query, modify, delete* the security attributes *[destination and encryption key]* to *[S.MANAGEMENT]*.

6.1.5.6 FMT_ MSA.3/SERVICES Static attribute initialisation

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

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

6.1.5.7 FMT_ MSA.3/CONFIG Static attribute initialisation

FMT_MSA.3.1/CONFIG The TSF shall enforce the *[Configuration Access Control SFP]* to provide *restrictive* default values for security attributes that are used to enforce the SFP.

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

6.1.5.8 FMT_ MSA.3/RED-BLACK Static attribute initialisation

FMT_MSA.3.1/RED-BLACK The TSF shall enforce the *[Red-black separation SFP]* to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/RED-BLACK The TSF shall allow the *[no one]* to specify alternative initial values to override the default values when an object or information is created.

6.1.5.9 FMT_MTD.1 Management of TSF data

FMT_MTD.1.1 The TSF shall restrict the ability to *modify, delete, [import]* the *[cryptographic keys]* to *[S.MANAGEMENT]*.

6.1.5.10 FMT_SMF.1 Specification of management functions

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

- *Traffic routes configuration*
- *Cryptographic keys configuration (Security Associations)*
- *Policies*
- *Radios configuration*

].

6.1.5.11 FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles *[S.MANAGEMENT, S.OPERATOR]*.

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

Application Note: The roles are implicitly assumed when:

- S.OPERATOR: Using TOE communication data capabilities.
- S.MANAGEMENT: Loading of configuration and key material from the non-volatile (internal and SD) memories during TOE boot and getting log files.

The S.MANAGEMENT is implicitly assumed when accessing the red network administration interface using the Management Centre. This role shall authenticate against Management Centre. The Management Centre will show the options corresponding to S.MANAGEMENT.

S.OPERATOR shall be able to user TOE communication data capabilities without authentication.

6.1.6 Protection of the TSF

6.1.6.1 FPT_FLS.1 Failure with preservation of secure state

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

– *self-test fails*

– *activation of tamper-response mechanisms*

– *activation of zeroization button*

].

Application Note: The preservation of the secure state requires the zeroization of the unprotected security parameters.

6.1.6.2 FPT_PHP.1 Passive detection of physical attack

FPT_PHP.1.1 The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

FPT_PHP.1.2 The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

Application Note: Security seals provide tamper evidence mechanisms for detecting unauthorized attempts to open the equipment.

6.1.6.3 FPT_PHP.3 Resistance to physical attack

FPT_PHP.3.1 The TSF shall resist *[unauthorized physical access attempt]* to the *[equipment inside]* by responding automatically such that the SFRs are always enforced.

Application Note: This SFR model the tamper-response mechanisms that zeroize the unprotected security parameters when the event of an unauthorized attempt to open the equipment is detected.

6.1.6.4 FPT_PHP.4 Opaque encapsulation

FPT_PHP.4.1 The TOE enclosure shall be opaque and physically continuous, so the direct observation or manipulation of inside the module is not possible.

6.1.6.5 FPT_STM.1 Reliable time stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

Aplication Note: Time stamps are generated by using a hardware component of the TOE.

6.1.6.6 FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_TDC.1.1 The TSF shall provide the capability to consistently interpret *[Configuration and cryptographic keys]* when shared between the TSF and another trusted IT product.

FPT_TDC.1.2 The TSF shall use *[mark language]* when interpreting the TSF data from another trusted IT product.

6.1.6.7 FPT_TST.2 TSF self-testing

FPT_TST.2.1 The TSF shall perform self-testing at power-up to verify the correctness of *[implemented encryption/decryption algorithms]* and of *[configuration data and X.509 certificates]* and to verify the integrity of the TSF-software/firmware.

FPT_TST.2.2 The TSF shall perform self-testing at the conditions *[at boot-time and every 5 minutes]* to verify the correctness of *[implemented encryption/decryption algorithms]*.

FPT_TST.2.3 The TSF shall perform self-testing at the conditions *[loading of new SW/FW]* to verify the correctness of *[none]* and to verify the integrity of *[the new SW/FW]*.

FPT_TST.2.4 The TSF shall perform self-testing at the conditions *[loading of configuration from the micro-SD]* to verify the integrity of *[configuration imported from the micro-SD card]*.

FPT_TST.2.5 The TSF shall provide *[Management role]* with the capability to invoke the following self-tests *[firmware integrity tests, configuration data integrity tests]*.

FPT_TST.2.6 After completion of self-testing the TSF shall *[output the results of self tests actions via the status output interface/port]*.

FPT_TST.2.7 If the **encryption/decryption algorithms** self-testing result is fail, the TSF shall *[zeroize and power off]*.

If the **firmware integrity** self-testing result is fail, the TSF shall *[zeroize and power off]*.

If the **configuration data integrity** self-testing result is fail, the TSF shall *[zeroize and power off, if the test fails three* *consecutive times]*.

Application Note: This SFR just outs the results of the self tests if they failed.

6.1.6.8 FPT_TOS.1 Trusted operating system

FPT_TOS.1.1 The underlying operating system must boot from a controlled source.

FPT_TOS.1.2 The underlying operating system integrity must be verified during boot.

FPT_TOS.1.3 The underlying operating system must have strong authentication mechanisms.

6.1.7 Trusted path/channels

6.1.7.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall be capable of using IPSec to provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit *the TSF, another trusted IT product* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for *[transfer of confidential data through the data interfaces]*.

6.1.7.2 FTP_TRP.1 Trusted path

FTP_ TRP.1.1 The TSF shall provide a communication path between itself and *remote* 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, disclosure*.

FTP_ TRP.1.2 The TSF shall permit *remote users* to initiate communication via the trusted path.

FTP_ TRP.1.3 The TSF shall require the use of the trusted path for *[communication with the Management Centre]*.

6.2 Security Assurance Requirements

The development and the evaluation of the TOE shall be done in accordance to the following security assurance requirements: EAL2 + ALC_FLR.1.

The following table shows the assurance requirements by reference to the individual components in [CC31R5P3]

Assurance Class	Assurance Components
ADV: Development	ADV_ARC.1: Security Architecture description ADV_FSP.2: Security-enforcing functional specification ADV_TDS.1: Basic design
AGD: Guidance documents	AGD_OPE.1: Operational user guidance AGD_PRE.1: Preparative procedures
ALC: Life-cycle support	ALC_CMC.2: Use of a CM system ALC_CMS.2: Parts of the TOE CM coverage ALC_DEL.1: Delivery procedures ALC_FLR.1: Basic flaw remediation
ASE: Security Target evaluation	ASE_CCL.1: Conformance claims ASE_ECD.1: Extended components definition ASE_INT.1: ST introduction ASE_OBJ.2: Security objectives ASE_REQ.2: Derived security requirements ASE_SPD.1: Security Problem Definition ASE_TSS.1: TOE summary specification
ATE: Tests	ATE_COV.1: Evidence of coverage ATE_FUN.1: Functional testing ATE_IND.2: Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2: Vulnerability analysis

Table 5 Security Assurance Requirements

6.3 Security Requirements Rationale

6.3.1 Necessity and sufficiency analysis

The following table presents a mapping of TOE Security Functional Requirements to Objectives.

TOE Security Objetive SFR	O.RED-BLACK	O.ROLES	O.AUDIT	O.SDCONFIG	O.DESTRUCTION	O.PHYSICAL	O.SELFTEST	O.MANAGEMENT	O.FIRMWARE	O.CRYPTO
FPT_TST.2							X		X	
FIA_509.1							X	X	X	
FIA_AFL.1								X		
FDP_UCT.1	X									
FDP_UIT.1	X									
FTP_ITC.1	X									
FDP_IFC.2	X									
FMT_MOF.1/AdminAct								X		
FMT_MTD.1		X		X				X		
FDP_ITC.2				X				X		
FPT_FLS.1					X					
FPT_PHP.3						X				
FPT_PHP.1						X				
FPT_PHP.4						X				
FAU_GEN.1			X							
FAU_SAR.1								X		
FCS_CKM.2										X
FCS_CKM.4					X					
FMT_SMR.1		X								
FPT_TOS.1										X
FAU_STG.2			X							
FAU_STG.4			X							
FDP_ACC.2/SERVICES		X						X	X	
FDP_ACF.1/SERVICES		X						X	X	
FMT_MSA.1/SERVICES		X						X	X	
FMT_MSA.3/SERVICES		X						X	X	

TOE Security Objetive / SFR	O.RED-BLACK	O.ROLES	O.AUDIT	O.SDCONFIG	O.DESTRUCTION	O.PHYSICAL	O.SELFTEST	O.MANAGEMENT	O.FIRMWARE	O.CRYPTO
FIA_509.2								X	X	
FIA_UAU.1		X								
FPT_STM.1			X							
FMT_SMF.1								X		
FDP_ACC.2/CONFIG		X		X				X		
FDP_ACF.1/CONFIG		X		X				X		
FMT_MSA.1/CONFIG		X		X				X		
FMT_MSA.3/CONFIG		X		X				X		
FPT_TDC.1								X		
FDP_IFF.1	X									
FCS_COP.1										X
FMT_MOF.1/TrustedUpdate									X	
FMT_MSA.1/RED-BLACK	X									
FMT_MSA.3/RED-BLACK	X									
FTP_TRP.1								X		

Table 6 SFRs/TOE Security Objectives coverage

6.3.2 Security Requirements Sufficiency

O.RED-BLACK: This security objective is mainly enforced by the Red-Black information flow control policy.

FDP_IFF.1, FDP_IFC.2, FMT_MSA.1/RED-BLACK FMT_MSA.3/RED-BLACK ensure that information travelling between red and black networks is encrypted/decrypted correctly.

FTP_ITC.1 requires the use of a trusted channel for data communications

FDP_UIT.1 and FDP_UCT.1 provide integrity and confidentiality of the data transmitted.

O.ROLES: **FMT_SMR.1** requires the TOE to provide different roles for the different available types of users.

The access control policy to the TOE services (**FDP_ACC.2/SERVICES, FDP_ACF.1/SERVICES, FMT_MSA.1/SERVICES, FMT_MSA.3/SERVICES**) describe which roles have access to which services.

The access control policy restrict access to encryption keys and other configurations to the S.MANAGEMENT role (**FDP_ACC.2/CONFIG, FDP_ACF.1/CONFIG, FMT_MSA.1/CONFIG, FMT_MSA.3/CONFIG, FMT_MTD.1**.

Roles are sometimes implicitly assumed as described in **FMT_SMR.1** and **FIA_UAU.1**.

O.AUDIT: The TOE generates audit logs (**FAU_GEN.1**) and stores it conveniently (**FAU_STG.2, FAU_STG.4**).

A reliable time of source (**FPT_STM.1**) is used to annotate the date of each log entry.

O.SDCONFIG: The configuration access control policy restrict access to encryption keys and other configurations to the S.MANAGEMENT role (**FDP_ACC.2/CONFIG, FDP_ACF.1/CONFIG, FMT_MSA.1/CONFIG, FMT_MSA.3/CONFIG, FMT_MTD.1**).

FDP_ITC.2 ensures that the imported configuration is encrypted.

O.DESTRUCTION: **FCS_CKM.4** provides method of destruction of stored CSPs.

The secure state will be preserved as is indicated in **FPT_FLS.1** when a destruction is done.

O.PHYSICAL: The security functional requirements **FPT_PHP.1**, **FPT_PHP.3** and **FPT_PHP.4** describes the physical capabilities implemented in the TOE in order to be resistant to physical attacks.

O.SELFTEST: The SFR **FPT_TST.2** describes the self-testing process implemented in the TOE.

It is complemented by **FIA_509.1** which describes specific measures for X.509 certificates.

O.MANAGEMENT: The TOE authenticates the management centre using X509 certificates (**FIA_509.1** and **FIA_509.2**).

This interface allows management of TOE functionality (**FMT_SMF.1, FMT_MOF.1/AdminAct, FMT_MTD.1**).

The access to this functionality is controlled by the services access control policy (**FDP_ACC.2/SERVICES, FDP_ACF.1/SERVICES, FMT_MSA.1/SERVICES, FMT_MSA.3/SERVICES**) and the configuration access control policy (**FDP_ACC.2/CONFIG, FDP_ACF.1/CONFIG, FMT_MSA.1/CONFIG, FMT_MSA.3/CONFIG**).

To use this interface, a trusted path must be opened by the client application (**FTP_TRP.1**).

Consistency in the configuration and cryptographic keys imported is ensured by **FPT_TDC.1** and **FDP_ITC.2**.

This interface is also used by the S.MANAGEMENT role to review logs (**FAU_SAR.1**).

If authentication fails repeatedly, the zeroization process will be invoked (**FIA_AFL.1**)

O.FIRMWARE: The services access control policy (**FDP_ACC.2/SERVICES, FDP_ACF.1/SERVICES, FMT_MSA.1/SERVICES, FMT_MSA.3/SERVICES**) makes this functionality only available to the S.MANAGEMENT role.

Upgrades authenticity and integrity is implemented through the **FMT_MOF.1/TrustedUpdate**, **FPT_TST.2**, **FIA_509.1** and **FIA_509.2** security functional requirements.

O.CRYPTO: **FCS_COP.1** and **FCS_CKM.2** implement this security objective requiring the TOE to implement cryptographic functionality.

It is used known standards with strong enough cryptographic key sizes.

The software implementation of cryptographic algorithms requires the use of a trusted operating system (**FPT_TOS.1**).

6.3.3 SFR Dependency Rationale

6.3.3.1 Table of SFR dependencies

The following table lists the dependencies for each requirement, indicating how they have been satisfied. The abbreviation "h.a." indicates that the dependency has been satisfied by a SFR that is hierarchically above the required dependency.

SFR	Required	Fulfilled	Missing
FPT_TST.2	FPT_FLS.1	FPT_FLS.1	None
FIA_509.1	None	None	None
FIA_AFL.1	FIA_UAU.1	FIA_UAU.1	None

SFR	Required	Fulfilled	Missing
FDP_UCT.1	[FTP_ITC.1 or FTP_TRP.1], [FDP_ACC.1 or FDP_IFC.1]	FTP_ITC.1, FDP_IFC.2 (h.a. FDP_IFC.1)	None
FDP_UIT.1	[FDP_ACC.1 or FDP_IFC.1], [FTP_ITC.1 or FTP_TRP.1]	FDP_IFC.2 (h.a. FDP_IFC.1), FTP_TRP.1	None
FTP_ITC.1	None	None	None
FDP_IFC.2	FDP_IFF.1	FDP_IFF.1	None
FMT_MOF.1/AdminAct	FMT_SMR.1, FMT_SMF.1	FMT_SMR.1, FMT_SMF.1	None
FMT_MTD.1	FMT_SMR.1, FMT_SMF.1	FMT_SMR.1, FMT_SMF.1	None
FDP_ITC.2	FPT_TDC.1, [FDP_ACC.1 or FDP_IFC.1], [FTP_ITC.1 or FTP_TRP.1]	FPT_TDC.1, FDP_ACC.2/CONFIG (h.a. FDP_ACC.1), FTP_TRP.1	None
FPT_FLS.1	None	None	None
FPT_PHP.3	None	None	None
FPT_PHP.1	None	None	None
FPT_PHP.4	None	None	None
FAU_GEN.1	FPT_STM.1	FPT_STM.1	None
FAU_SAR.1	FAU_GEN.1	FAU_GEN.1	None
FCS_CKM.2	[FCS_CKM1 or FDP_ITC.1 or FDP_ITC.2], FCS_CKM.4	FDP_ITC.2, FCS_CKM.4	None
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FDP_ITC.2	None
FMT_SMR.1	FIA_UID.1	None	FIA_UID.1
FPT_TOS.1	None	None	None
FAU_STG.2	FAU_GEN.1	FAU_GEN.1	None
FAU_STG.4	FAU_STG.1	FAU_STG.2 (h.a. FAU_STG.1)	None
FDP_ACC.2/SERVICES	FDP_ACF.1	FDP_ACF.1/SERVICES	None
FDP_ACF.1/SERVICES	FDP_ACC.1, FMT_MSA.3	FDP_ACC.2/SERVICES (h.a. FDP_ACC.1), FMT_MSA.3/SERVICES	None
FMT_MSA.1/SERVICES	FMT_SMR.1, FMT_SMF.1, [FDP_ACC.1 or FDP_IFC.1]	FMT_SMR.1, FMT_SMF.1, FDP_ACC.2/SERVICES (h.a. FDP_ACC.1)	None
FMT_MSA.3/SERVICES	FMT_MSA.1, FMT_SMR.1	FMT_MSA.1/SERVICES, FMT_SMR.1	None
FIA_509.2	None	None	None
FIA_UAU.1	FIA_UID.1	None	FIA_UID.1
FPT_STM.1	None	None	None

SFR	Required	Fulfilled	Missing
FMT_SMF.1	None	None	None
FDP_ACC.2/CONFIG	FDP_ACF.1	FDP_ACF.1/CONFIG	None
FDP_ACF.1/CONFIG	FDP_ACC.1, FMT_MSA.3	FDP_ACC.2/CONFIG (h.a. FDP_ACC.1), FMT_MSA.3/CONFIG	None
FMT_MSA.1/CONFIG	FMT_SMR.1, FMT_SMF.1, [FDP_ACC.1 or FDP_IFC.1]	FMT_SMR.1, FMT_SMF.1, FDP_ACC.2/CONFIG (h.a. FDP_ACC.1)	None
FMT_MSA.3/CONFIG	FMT_MSA.1, FMT_SMR.1	FMT_MSA.1/CONFIG, FMT_SMR.1	None
FPT_TDC.1	None	None	None
FDP_IFF.1	FDP_IFC.1, FMT_MSA.3	FDP_IFC.2 (h.a. FDP_IFC.1), FMT_MSA.3/RED-BLACK	None
FCS_COP.1	FCS_CKM.4, [FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.4, FDP_ITC.2	None
FMT_MOF.1/TrustedUpdate	FMT_SMR.1, FMT_SMF.1	FMT_SMR.1, FMT_SMF.1	None
FMT_MSA.1/RED-BLACK	FMT_SMR.1, FMT_SMF.1, [FDP_ACC.1 or FDP_IFC.1]	FMT_SMR.1, FMT_SMF.1, FDP_IFC.2 (h.a. FDP_IFC.1)	None
FMT_MSA.3/RED-BLACK	FMT_MSA.1, FMT_SMR.1	FMT_MSA.1/RED-BLACK, FMT_SMR.1	None
FTP_TRP.1	None	None	None

Table 7 SFR Dependencies

6.3.3.2 Justification for missing dependencies

FMT_SMR.1 dependency on FIA_UID.1

The dependency is not satisfied because the TOE does not need identification of the user S.OPERATOR to work without management capabilities.

FIA_UAU.1 dependency on FIA_UID.1

The dependency is not satisfied because the TOE does not need identification of the user.

6.3.4 SAR Rationale

The selected SAR package is selected to fulfil the required market level for this kind of products.

6.3.5 SAR Dependency Rationale

6.3.5.1 Table of SAR dependencies

SAR	Required	Fulfilled	Missing
ASE_CCL.1	ASE_INT.1, ASE_ECD.1, ASE_REQ.1	ASE_INT.1, ASE_ECD.1, ASE_REQ.2 (hierarchically above ASE_REQ.1)	None
ASE_ECD.1	None	None	None
ASE_INT.1	None	None	None
ASE_OBJ.2	ASE_SPD.1	ASE_SPD.1	None
ASE_REQ.2	ASE_OBJ.2, ASE_ECD.1	ASE_OBJ.2, ASE_ECD.1	None
ASE_TSS.1	ASE_INT.1, ASE_REQ.1, ADV_FSP.1	ASE_INT.1, ASE_REQ.2 (hierarchically above ASE_REQ.1), ADV_FSP.2 (hierarchically above ADV_FSP.1)	None
ALC_CMC.2	ALC_CMS.1	ALC_CMS.2 (hierarchically above ALC_CMS.1)	None
ALC_CMS.2	None	None	None
ADV_FSP.2	ADV_TDS.1	ADV_TDS.1	None
AGD_OPE.1	ADV_FSP.1	ADV_FSP.2 (hierarchically above ADV_FSP.1)	None
AGD_PRE.1	None	None	None
ATE_IND.2	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1, ATE_COV.1, ATE_FUN.1	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1, ATE_COV.1, ATE_FUN.1	None
AVA_VAN.2	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1	None
ASE_SPD.1	None	None	None
ALC_DEL.1	None	None	None
ADV_ARC.1	ADV_FSP.1, ADV_TDS.1	ADV_FSP.2 (hierarchically above ADV_FSP.1), ADV_TDS.1	None
ADV_TDS.1	ADV_FSP.2	ADV_FSP.2	None
ATE_COV.1	ADV_FSP.2, ATE_FUN.1	ADV_FSP.2, ATE_FUN.1	None
ATE_FUN.1	ATE_COV.1	ATE_COV.1	None
ALC_FLR.1	None	None	None

Table 8 SAR dependencies

7 TOE SUMMARY SPECIFICATION

7.1 TOE Summary

SFR	Summary
FDP_IFC.2 ***FDP_IFF.1*** ***FMT_MSA.1/RED-BLACK*** ***FMT_MSA.3/RED-BLACK*** ***FDP_UIT.1*** ***FDP_UCT.1***	These SFRs prove the red-black separation. The TOE has been designed to have two clearly divided zones: red and black. The TOE has SFP to control the information flow between the red and black zone **(FDP_IFC.2, FDP_IFF.1)** that allow: – Configuration in the red zone. – Send data from the red to the black zone with the S.OPERATOR role. – Receive data from the black to the red zone with the S.OPERATOR role. The device, physically speaking, is compound of two independent boards where each zone (red and black) runs in an isolated way. At the initialisation of the device **(FMT_MSA.1/RED-BLACK, FMT_MSA.3/RED-BLACK)**: 1. External interfaces of both red and black zones are disabled. 2. Internal communication buses between black zone and red zone are configured. 3. The red zone decrypts the black zone configuration data. 4. This configuration data is sent to the black zone through internal buses. 5. Black zone configures itself. 6. The internal communication buses are disabled. 7. External interfaces of both red and black zones are enabled. The outgoing data from the red zone is ciphered. Likewise, the ingoing data to the red zone must be ciphered **(FDP_UCT.1, FDP_UIT.1)**.

SFR	Summary
	Therefore, the ingoing and outgoing data to and from the black zone are ciphered. The TOE uses the IPSec protocol to assure **(FDP_UCT.1, FDP_UIT.1)**: – Confidentiality. – Data integrity.
FTP_ITC.1 ***FTP_ITC.2*** ***FIA_509.1*** ***FCS_COP.1***	These SFRs prove the channel is trusted **(FTP_ITC.1, FTP_ITC.2)**. These communication channels are logically distinct from other communication channels, providing assured identification of its end points **(FIA_509.1)** and protection of the channel data from modification or disclosure. The following protocols are used: – Data: IPSec protocol **(FCS_COP.1)**. o IPSec tunnel in transport or tunnel mode, between the end points of the red zone.
FDP_ACC.2/SERVICES ***FDP_ACF.1/SERVICES*** ***FMT_MSA.1/SERVICES*** ***FMT_MSA.3/SERVICES*** ***FMT_SMR.1*** ***FIA_UAU.1***	These SFRs prove the role implementation. The TOE implements the following roles: – S.MANAGEMENT – S.OPERATOR The TOE, at booting, establishes the accessible services by the role using the TOE, and configures it based on its role **(FMT_SMR.1, FIA_UAU.1)**. The user will access the following functions based on its role **(FDP_ACC.2/SERVICES, FDP_ACF.1/SERVICES)**: – S.MANAGEMENT: o System booting o Initialization o Key import

<table>
<tr><th>SFR</th><th>Summary</th></tr>
<tr><td></td><td>
<ul><li>IPSec configuration</li><li>Software Update</li><li>Logs auditing.</li></ul>
– S.OPERATOR:
<ul><li>Encryption and decryption of data.</li></ul>
<p>The following table indicates the accessible services by the user based on its role <b>(FMT_MSA.1/SERVICES, FMT_MSA.3/SERVICES)</b>:</p>
<table>
<tr><th>Service</th><th>Roles</th></tr>
<tr><td>Show status</td><td>Any</td></tr>
<tr><td>Key loading</td><td>S.MANAGEMENT</td></tr>
<tr><td>Configuration</td><td>S.MANAGEMENT</td></tr>
<tr><td>Encrypt/Decrypt data</td><td>S.OPERATOR</td></tr>
<tr><td>Audit review</td><td>S.MANAGEMENT</td></tr>
<tr><td>System Maintenance & FW/SW upgrade</td><td>S.MANAGEMENT</td></tr>
</table>
</td></tr>
<tr><td><b><i>FAU_GEN.1</i></b>
<b><i>FAU_SAR.1</i></b>
<b><i>FAU_STG.2</i></b>
<b><i>FAU_STG.4</i></b>
<b><i>FPT_STM.1</i></b>
<b><i>FCS_COP.1</i></b></td><td>
<p>These SFRs prove the audit.</p>
<p>The TOE makes the following operations when the Security supervision of the system starts <b>(FCS_COP.1, FAU_STG.2, FAU_STG.4)</b>:</p>
<ol><li>Decrypt the log file located in the non-volatile memory</li><li>Check the information integrity</li></ol>
<p>The TOE makes the following operations when a new event mentioned in the SFRs is recorded <b>(FAU_GEN.1, FPT_STM.1)</b>:</p>
<ul><li>Record a new event at the end of the log file.</li></ul>
<p>The TOE makes the following operations when system is being shutting down <b>(FAU_STG.2, FAU_STG.4)</b>:</p>
<ol><li>Create a new hash with the modified log file.</li><li>Encrypt the modified log file <b>(FCS_COP.1)</b>.</li></ol>
<p>The TOE audit log is accessed by the user with the role S.MANAGEMENT from the Management</p>
</td></tr>
</table>

SFR	Summary
	Centre, which process it and shows in an appropriate manner **(FAU_SAR.1)**. The TOE has several security measures related to record the events **(FAU_STG.2, FAU_STG.4)**: – The system only starts if there is enough memory to save the system events. The free memory capacity has to be, at least, of 25% of the designated memory. – The TOE check the log file size when a new event is recorded. If the maximum size is reached, first events recored are erased to achieve that to register new events. The TOE has been design keeping a security and operability compromise in order to guarantee the communication. Therefore, some concession has been made since non-connectivity could be an operational risk. Following this policy, the TOE does not zeroize the system if the maximum memory size is reached to record log events and the equipment is allowed to continue running.
FDP_ACC.2/CONFIG ***FDP_ACF.1/CONFIG*** ***FMT_MSA.1/CONFIG*** ***FMT_MSA.3/CONFIG*** ***FMT_MTD.1*** ***FDP_ITC.1*** ***FDP_ITC.2*** ***FIA_AFL.1*** ***FCS_CKM.2*** ***FCS_COP.1*** ***FIA_509.1***	These SFRs prove the configuration from the micro-SD. The TOE, when it is powered up, initiates the process of initialization and configuration of the TOE services. It does the next steps: – Booting: The OS and application are decrypted and validated from the hardware. – The application decrypts and validates the information stored in the micro-SD. It configures the security functions from the information located in the micro-SD card **(FCS_COP.1)**.

SFR	Summary
FIA_509.2	– The application is verified with a certificate contained in micro-SD **(FIA_509.1, FIA_509.2)**. – The X.509v3 certificate of micro-SD is validated **(FIA_509.1, FIA_509.2)**. – Cryptographic keys are imported and IPSec is configured. – Depending on whether Management Centre is connected **(FDP_ACC.2/SERVICES, FDP_ACF.1/CONFIG, FMT_MSA.1/CONFIG, FMT_MSA.3/CONFIG)**: o If Management Centre is connected and S.MANAGEMENT role stands, TOE can reach management capabilities. o If Management Centre is not connected, S.OPERATOR role capabilities are configured in TOE. If during TOE boot, several failed attempts for validating the micro-SD card are detected, the system is zeroized **(FIA_AFL.1)**. The maximum number of allowed failed attempts is 3 (it can be customized). Once the device has been configured by using the S.MANAGEMENT role capabilities, this role is abandoned and the TOE is identified with S.OPERATOR role, so the following operations can be performed: • Encrypt/Decrypt Data with IPSec protocol ***(FCS_COP.1)*****.** The Management Centre can connect to the TOE through the management interface **(FTP_ITC.1, FTP_ITC.2)**. Once it is connected, a properly command is sent to the TOE in order to start the S.MANAGEMENT role capabilities **(FMT_MTD.1, FTP_ITC.2, FDP_ACC.2/SERVICES,**

SFR	Summary
	FDP_ACF.1/CONFIG, FMT_MSA.1/CONFIG, FMT_MSA.3/CONFIG). The TOE, once the Management Centre has been identified **(FDP_ITC.1, FDP_ITC2)**: – Block all black zone interfaces. – Enable sockets to receive commands from Management Centre. At this moment, the S.MANAGEMENT role thought Management Centre can **(FDP_ACC.2/SERVICES, FDP_ACF.1/CONFIG, FMT_MSA.1/CONFIG, FMT_MSA.3/CONFIG)**: – Retrieve TOE log files. – Update mission and cryptographic keys in the micro-SD – Perform the re-key operation – Update TOE Software In order to import data created outside of the TOE, it must apply the following control rules: – Decrypt and validate the signature associated with this data **(FCS_COP.1)**. – Check the X.509 certificate included in the micro-SD is correct **(FIA_509.1, FIA_509.2)**. The TOE only import encrypt and protected with signature data through the secure channel established with Management Centre **(FDP_ITC.1, FDP_ITC.2)**.
FPT_PHP.1 ***FPT_PHP.3*** ***FPT_PHP.4*** ***FPT_FLS.1***	This SFRs proves the physic protection of the TOE. The TOE is protected for a continue and opaque box **(FPT_PHP.4)**. The TOE has open switches and tamper evidence seals to detect if anyone has been trying to open the device **(FPT_PHP.3)**. It also detects temperature and voltage attacks **(FPT_PHP.1)**. It means if the current environmental temperature does not fit to

SFR	Summary
	a normal operative range, or the voltage is outside of the normal operative range, thus an attack is considered. If the TOE detects an access try to the equipment, the system is zeroized **(FPT_FLS.1)**.
FPT_TST.2 ***FIA_509.1*** ***FIA_509.2*** ***FPT_FLS.1*** ***FDP_ACC.2/SERVICES*** ***FDP_ACF.1/SERVICES*** ***FMT_MSA.1/SERVICES*** ***FMT_MSA.3/SERVICES*** ***FMT_MOF.1/TrustedUpdate***	This SFRs prove the necessary functions to check the system integrity, tests make by the TOE, erase classified material and OS and software update. The TOE make several kind of tests **(FPT_TST.2)**: – Start-up test. – Periodic test (every 5 minutes). – Requested user test. The tasks when the system start-up are **(FPT_TST.2)**: – OS Integrity and signature (critic). – Application software Integrity (critic). – Check if there is a micro-SD inserted (critic if tries number reachs three times). – Check the information integrity stored in the micro-SD (critic). – Check the quantity of available non-volatile memory (not critic). – Check system vulnerabilities: open switches, temperature and voltage. – Check the X.509 certificate validity and if it is having been issued by a properly CA (critic) **(FIA_509.1, FIA_509.2)**. – Check if the serial number located in the X.509 certificate is right (critic) **(FIA_509.1, FIA_509.2)**. – Cryptographic algorithms. Periodic tests (CBIT) are made every 5 minutes to check: – Cryptographic algorithms (critic). – Physical interfaces (not critic).

SFR	Summary
	The TOE block the output of interfaces/ports when the following tests are in progress: – Initial start-up. – Self-test. – Power-up self-test. – Self-test at the request of authorized user. The TOE update the OS and software via Management Centre **(FMT_MOF.1/TrustedUpdate)**. Once the TOE has been initiated, a subject with S.MANAGEMENT role can connect the Management Centre to the red zone of the TOE and update the OS and the software **(FDP_ACC.2/SERVICES, FDP_ACF.1/SERVICES, FMT_MSA.1/SERVICES, FMT_MSA:3/SERVICES).** If any error is detected in errors tagged as (*critic),* TOE zeroizes CSP data **(FPT_FLS.1)**.
FIA_509.1 ***FIA_509.2*** ***FMT_SMF.1*** ***FMT_MOF.1/AdminAct*** ***FTP_TRP.1*** ***FPT_TDC.1***	This SFRs prove the configuration functions of the TOE. Once the TOE has started, the Management Centre can connect with the TOE **(FPT_TDC.1)**. In the same way, the TOE also use X.509 certificates in order to authenticate the Management Centre **(FIA_509.1, FIA_509.2, FTP_TRP.1)**. The S.MANAGEMENT role establishes the Management functions and security policies of the TOE to enable and config security functions **(FMT_MOF.1/AdminAct, FMT_SMF.1)**.

SFR	Summary
FCS_COP.1 ***FPT_TOS.1*** ***FCS_CKM.2*** ***FCS_CKM.4*** ***FIA_509.1*** ***FIA_509.2***	This SFRs prove the cryptographic algorithms and cryptographic operations made for the TOE. The TOE uses standard algorithms to encrypt and digest data **(FCS_COP.1)**: – AES 256 in modes ECB, GCM, CCM and CBC – RSA 2048. – SHA 256. – ECDSA within Brainpool 384 – PBKDF2 Once the TOE has obtained **(FCS_CKM.2)** and used the keys, to decrypt the information located in the non-volatile memories (internal and micro-SD storage), erase them from the system non-volatile memory. The erase method is to write zeros in the volatile memory of the key **(FCS_CKM.4)**. The TOE hardware is whose decrypt and check the signature of u-boot and linux kernel on Secure Boot process **(FPT_TOS.1, FCS_COP.1)**. Both OS are stored in the non-volatile memory of the TOE. Furthermore, it is guarantee the correct implementation of re-key operation **(FCS_CKM.2)**, providing a secure way to update TOE keys (IPSec), and verifying and authenticating these packets **(FIA_509.1, FIA_509.2)**.

Table 9 TOE Summary Specification

END OF DOCUMENT