Färist IEG - Security Target

Document version: 1.0

Date: 12/02/26

Title	Färist IEG - Security Target		
Version:	1.0	Developer:	Tutus Data AB
Status:	Released	Classification:	Public

Tutus Data AB, Svärdvägen 11, SE-182 33 Danderyd, Tel: +46 (0)8 551 102 30 Fax: +46 (0)8 551 102 39, E-mail: info@tutus.se, Web site: http://www.tutus.se

Document history

Version	Date	Author	Changes to previous version
1.0	2026-02-11	Rickard Lind	Initial released version

Table of contents

Index of tables

Index of illustrations

1 Introduction

1.1 Security Target Identification and organisation

Title:	Färist IEG - Security Target
Version:	1.0
Status:	Released
Date:	12/02/26
Sponsor:	Tutus Data AB
Developer:	Tutus Data AB
Keywords:	Security Target, Common Criteria, Tutus, Virtual Private Network, VPN, Firewall, Networking, IEG, Information Exchange Gateway, Access Control

This ST has been structured in accordance with [CC] Part 1. The main sections of the ST are the introduction, security problem definition, security objectives, security requirements, TOE summary description and annexes.

The introduction provides general information about the TOE, serves as an aid to understand the nature of the TOE and its security functionality, and provides context for the ST's evaluation.

The security problem definition describes the security aspects of the environment in which the TOE is to be used and the manner in which it is to be employed. The TOE security environment includes:

a) assumptions regarding the TOE's intended usage and environment of use

b) threats relevant to secure TOE operation

c) organisational security policies with which the TOE must comply

The security objectives reflect the stated intent of the ST. They pertain to how the TOE will counter identified threats and how it will cover identified organisational security policies and assumptions. Each security objective is categorised as being for the TOE or for the environment. The security objectives rationale demonstrates that the stated security objectives are traceable to all of the aspects identified in the TOE security problem definition and that they are suitable to cover them.

The security requirements section provides detailed requirements, in separate subsections, for the TOE and its environment. The security requirements are further divided into the TOE security functional requirements and the TOE security assurance requirements.

The TOE summary specification addresses the security functions that are represented by the TOE to answer the security requirements.

The annex contains a list of abbreviations and a glossary relevant for this ST.

1.2 TOE Identification

The TOE is the firmware part of Färist IEG version 4.5.0.

1.3 TOE Type

The TOE type is a networking device. In particular, the TOE covered by this ST is an IEG (Information Exchange Gateway). It consists of an information filtering system with interchangeable filter modules. It also uses VPN (Virtual Private Network) functionality.

1.4 TOE Overview

The TOE is an IEG that implements an information exchange policy between a *source* and a *destination* network, where *source* is the network initiating a session. The networks may be *local* or *remote*. Local networks are defined by TOE network interfaces. Remote networks are defined by authenticated VPN tunnels.

The policy is implemented using interchangeable filter modules. The TOE is delivered with two filter modules:

- NTP-filter: Enables time synchronization with minimal information leakage. The leakage from *destination* to *source* is limited to 2500 bits over each 5 minute time slot.
- UDP-diode: Only allows traffic in one direction, from *source* to *destination*.

Additional filter modules can be loaded onto the device to handle other filtering requirements. There are also two built-in filter modules, *Allow* (which passes all packets unmodified) and *Deny* (which drops all packets).

The base system is configured with filter rules that allow IP-traffic to and from the filter module (layer 3 and 4) and the filter module handles any kind of filtering above that. The term *filter* refers to the filter module and rules taken together.

The TOE also also includes the following security functions:

- Audit: The TOE maintains a log, the purpose of which is traceability and records of events like startup, reconfiguration and similar. It also records security related events like attempts to connect using wrong/old credentials, integrity error on packets, internal integrity control and similar. All log events are sent to a configured log server. The TOE supports different log levels (severities) of auditing. In the evaluated configuration the log level “info” has to be used.
- Administration capabilities: An HTTPS API is offered to administrators for all relevant configuration of security functionality. There is also a front panel for displaying status information. The HTTPS API uses a certificate based authentication. When using the HTTPS API there are two roles, one administrator with full privileges, and one that can only read information (operator). Things that can be written include the configuration, certificate, CRLs, filters and system updates. Things that can be read includes configuration, run time status, statistics and the locally saved log.
- Autoupdate: The TOE can be configured to automatically check for new firmware versions on a specified server and update itself. The integrity and authenticity of the updates will be verified. Only updates to newer (higher) versions can be done.
- Selftest and failsafe: The TOE also has selftest and failsafe capabilities.

1.5 TOE Description

1.5.1 Introduction

This document describes the architecture of the Färist IEG. The TOE is an IEG device that will enforce a communication policy between two or more networks, which means it will function as an IP Router. The networks may be local (on an interface) or remote. Remote networks are accessed using IPSEC VPN functionality where encryption is done on IP-packets and the encrypted packets are transmitted according to the IPSec standard.

1.5.2 Intended use

The intended use of the TOE is to connect networks that may have different provenance and have it enforce a communications policy between them. The policy is enacted by filtering of traffic between the networks.

To operate the TOE securely the operating environment must:

- Physically protect the TOE from unauthorised access at all times
- Contain a log server
- Have the ability to generate certificates of high quality

It is recommended that the environment also contain:

- An NTP-server
- An administrative client

1.5.3 Architecture

1.5.3.1 Overview

The TOE consist of two subsystems, the control plane and the data plane. The control plane runs on top of a Linux kernel while the data plane operates directly on top of the hardware. All packets are received and validated by the data plane. The dataplane routes packets according to a rule table. Packets directed to the TOE itself or a via a filter are sent to the control plane. Packets to or from a VPN tunnel are sent to the crypto engine. Packets that do not match any rule are dropped.

The components of all subsystems are described below.

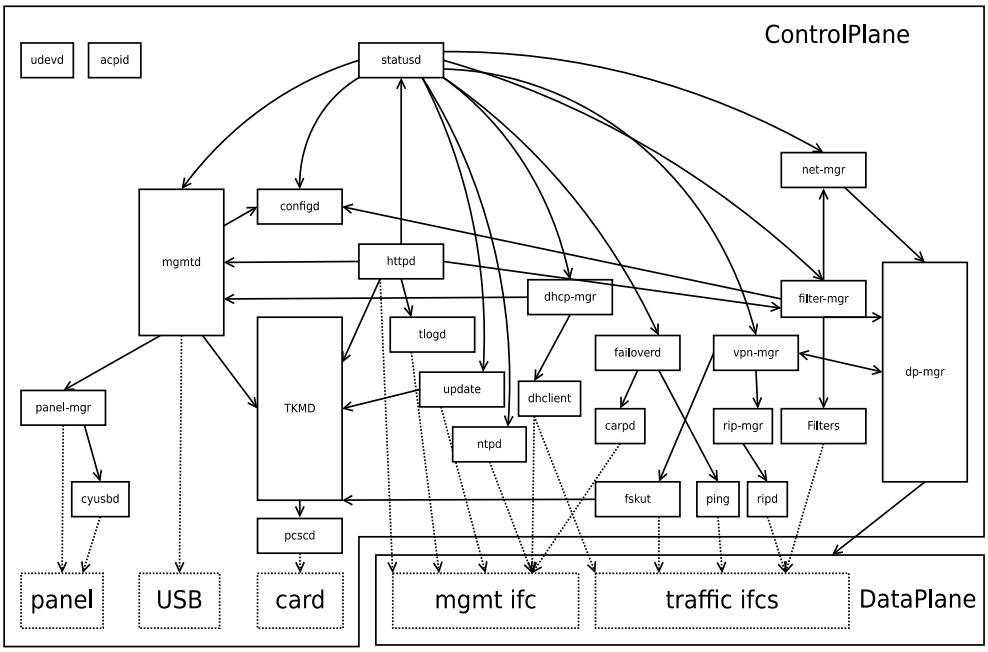

Illustration 1: Färist IEG Architecture

1.5.4 Hardware

Illustration 2: Färist IEG – H220 and M110

In the evaluated configuration, the TOE can run on two hardware platforms, one office device and one semi-rugged device. Both devices have a serial console, dedicated Ethernet ports and a front panel for management. The H220 has 16 traffic ports of 1Gb/s and 4 ports of 10 Gb/s while the M110 has 4 traffic ports of 1Gb/s. The configuration is stored on a small USB memory that must be present at all times while the system is running. The keys can either be stored on the USB-memory or on a Smart Card.

1.5.5 Control Plane

The control plane handles all administrative roles in the TOE, including key negotiations. It consists of a number of components which run on top of a standard Linux kernel.

The components of the control plane subsystem are listed and described below:

mgmtd	Management service – bootstraps and monitors the software.
fsintegrity	Service that verifies file system integrity .
configd	Configuration service – configures the software.
config file	The configuration file describes all configuration options for the whole TOE. It resides on a removable USB memory and holds the entire configuration state of the TOE. The config file can also contain CA-certificates.
dp-mgr	DataPlane manager – bootstraps the data plane and translates commands exchanged between control plane and data plane. The exchange is done over a local socket.
net-mgr	Network manager – manages all IP configuration in the system, including setup interface addresses and routes in control plane and sending commands with respective IP configuration data to data plane.
filter-mgr	Filter manager – manages filter modules and updates data plane rule table.
vpn-mgr	VPN manager – manages VPN tunnels.
update	Downloads CRLs, filter and system update packages. It will connect to configured update server and request a new version of the software. It verifies the digital signature of the update using a factory installed update signing certificate and also that the software version in the update is newer than the running version.
statusd	Status service – collects and makes available statistics from all other components.
panel-mgr	Panel manager – prints out statistics and machine status on the front panel. Can also accept administrative commands from the front panel.
cyusbd	**M110 only**: driver for the panel, keypad and network traffic indicators.
rip-mgr	Announces routes using RIPv2.
snmp	SNMP agent that provides select status.
symvpnd	Manager for VPN tunnels with pre-shared symmetrical keys (unicast and multicast).
tkmd	This service provides signing with the private key of the Färist IEG. Signing is done using a key stored on the USB-memory or a Smart Card. Collects entropy thorough various means in the system and keeps an entropy state on disk. Serves random seed to components requesting it through a Unix socket.
pcscd	This service manages the smart card.
fskut	Negotiates keys with remote peers.
tlogd	Log daemon, accepts log messages from all components and sends them to a remote log server.
ntpd	Adjust system time against a remote NTP server.

dhcp	The DHCP subsystem. It can negotiate addresses for the interfaces as a client.
failoverd	Provides high availability with failover between two units.
carpd	Communicates with failover peer using multicast on management network.
ping	Verifies connectivity on traffic interfaces.
udevd	Device management daemon.
acpid	Power management daemon.
httpd	Remote administration daemon exposes a simple management interface to an authenticated administrator. The supported commands include: • Get configuration • Save configuration • Activate configuration • Get status of the TOE with traffic load and similar information The commands are in form of HTTP requests/responses. The administration is done through a separate management client.
Keys	Asymmetric keys stored in a PKCS#12 archive. The keys are either stored on the same USB memory that holds the configuration file or they are stored on an external smart card. The asymmetric keys are only used for key negotiation, all encryption is done employing AES-256 symmetric encryption algorithm with the negotiated keys.
Linux kernel	Standard Linux kernel with special drivers to manage the data plane and receive and send IP-packets to it.

1.5.6 Data Plane

The data plane is running directly on the hardware and is responsible for all packet processing. The data plane consists of two modules which are listed and described below:

Flow Multiplexor	Performs packets parsing, basic validation and filtering. Based on the rule table a packet can be sent to the control plane, a filter or the VPN module for encryption/decryption.
VPN	Traffic if configured can be sent to VPN module for encryption/decryption. The interface is packet queue based and can be used for exchanging packets in both directions.

1.5.7 Administration

The TOE requires a configuration file and keys in order to work as intended. Administration is limited to manipulating configuration files and keys and to obtain output of status and log events.

Administration can be done using two different interfaces:

- **Front panel:** The front panel can be used to see the running status of

the TOE and perform some simple administrative tasks.

- **Remote administration:** Using a separate administration client the TOE can be remotely administrated through an authenticated TLS interface.

In addition, the USB interface is used by the administrator to import management data into the TOE.

Different administrative functions are available on these interfaces. The front panel can be used to view the status of the running device as well as performing simple administrative tasks like shutting down the device.

Remote administration can be used to view status of the device, update the configuration and read the log.

1.5.8 Physical scope of the TOE

The TOE is software only. It consists of the Färist IEG firmware version 4.5.0.

Relevant guidance documents for the secure operation of the Färist IEG that are part of the TOE are:

- Administrator's Reference Manual Färist IEG 4.5.0

Physical boundaries between the TOE and its runtime environment are described below:

- The base Linux operating system, other than particular applications implementing TSF, is considered part of the runtime environment.
- The hardware is considered part of the runtime environment. The following TOE and hardware appliance model combinations are covered by this evaluation:
  - Färist IEG running on H220
  - Färist IEG running on M110

The following components can be found in the operational environment of the TOE on systems other than those hosting the TOE:

- Client software, the administrative client is not part of the TOE.
- Log servers (e.g. syslog server and analysis tools used for collecting the audit records and for their analysis) and NTP servers.

1.5.9 Evaluated configuration

The evaluated configuration requires the following restrictions:

- A remote log server must be configured with the **Level** option set to *Info*.
- Remote administration must require TLS 1.3 by setting the **AllowLegacyTLS** option to *false* in the System Administration section.
- The Symmetrical VPN and Multicast functions (*SymVPN* and *Multicast* in the configuration) may not be used.
- VPN when used must require post-quantum ready encryption by setting the **PQ** option to *Mandatory*.
- Only the UDP-diode and NTP-filter filter modules may be used; no other fil-

ters may be configured in the *Filter* section of the configuration.

1.5.10 Logical scope of the TOE

This section provides an overview of the security functions implemented by the TOE.

1.5.10.1 Filter

The filter functionality makes it possible to enforce basically any communication policy between two or more networks. Filter modules are programs that are loaded to the device and are configured using the configuration file.

The base system will direct traffic based on IP and TCP/UDP information to the filter program that performs the filtering.

Filters for NTP and a UDP-diode are included in the evaluation.

The NTP filter allows time synchronization using the NTP protocol while limiting the information leakage.

The UDP diode forwards UDP packets in one direction and drops any response packets in the other.

1.5.10.2 VPN

The VPN functionality uses encryption for the authentication and integrity of remote networks.

Key exchange and authentication is done with the SKUT protocol [SKUT5] using an RSA-signed Diffie-Hellman key exchange combined with the FrodoKEM (eFrodoKEM-1344-SHAKE) key encapsulation to make it quantum-resistant. Encryption is done using IPSec ESP [RFC4303] in tunnel mode with the AES-256-GCM encryption algorithm.

VPN tunnels are configured using the configuration file and the TLS certificate stored in a file on USB memory or on a smart card.

1.5.10.3 Audit

Audit records that are created in different components are sent to a log daemon (tlogd), that will forward the audit records to one or more remote log servers.

If the log servers are not reachable the log will be kept in RAM until it can be sent.

The last few log events are also stored in a local copy of the audit file, which is kept in RAM. This is used for troubleshooting and is not considered a TSF.

1.5.10.4 Management

The TOE can have the following administrative roles:

- Administrator
  - Remote administrator
    The remote administrator can perform all administrative tasks, including changing keys.
    The remote administrator is authenticated through their certificate.
  - Local administrator
    The local administrator can perform the same tasks as the remote

administrator. In addition, the local administrator can change the smart card.
The local administrator is authenticated through organizational means, by allowing only authorized personnel physical access to the TOE.

- Operator
  The (remote) operator can perform the same tasks as the remote administrator but cannot perform any changes, such as changing configuration. The operator is authenticated through their certificate.

Remote management is secured through certificates. The remote user is identified through the CN (common name), UID (Unique Identifier) or Email (RFC822 email) field in their certificate.

Local management is secured through organizational means and is therefore not considered a TSF.

1.5.10.5 Autoupdate

The TOE has an automatic update functionality that checks for new firmware updates, verifies the origin and integrity of the update and ensures that the update is newer than the current version. Note that it is the update function that is part of the TOE and not the updated version of the TOE.

1.5.10.6 Selftest and failsafe

The TOE has built-in functionality for self tests that are run both at startup and at regular intervals. Self tests verify the integrity of the system files and ensure the proper working of the encryption engine. If a self test fails the TOE will restart.

If the USB-memory or smart card holding the TLS certificate/private key is removed the TOE will also restart.

1.5.11 Operational environment support

1.5.11.1 Physical environment

The TOE must be protected against unauthorised access at all times.

1.5.11.2 IT environment

The IT environment **must** contain the following:

- Log server
- CA – certificate generation

The IT-environment **may** contain the following:

- NTP server. An NTP server on the administrative network can be used by the TOE to set correct time.
- Administrative client

1.5.12 Secure delivery

The TOE may only be delivered to the end customer by approved transporters. The TOE chassis is fitted with seals that make tampering evident. The receiver is instructed to reject the delivery if a seal is broken or its serial number does not match the manifest.

2 CC Conformance Claim

This ST is CC Part 2 extended and CC Part 3 conformant, with a claimed Evaluation Assurance Level of EAL4, augmented by ALC_FLR.1.

This ST does not claim conformance to any Protection Profile.

Common Criteria [CC] version 2022 release 1 is the basis for this conformance claim.

This conformance claim also takes into account the Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1) [CCERR].

3 Security Problem Definition

The TOE enforces a communication policy between two or more networks by operating as an IP router. In this context, networks are defined as follows:

- **Local networks:** Directly connected to the TOE via its network interfaces.
- **Remote networks:** Accessed over authenticated IPSec VPN tunnels, where IP packets are encrypted and transmitted in accordance with the IPSec standard.

The policy consists of traffic filtering with filters that have the ability to completely control the traffic that passes between networks. The filters are modular and can be installed, configured and updated by the administrator. The policy covers all traffic combinations: local to local, local to remote, remote to local, and remote to remote.

Note that the VPN is used solely to verify that traffic originates from a remote network by providing cryptographic authentication and integrity, not as an access control mechanism.

The TOE is not only in itself an asset, but it aims to protect assets which are in the networks to which it is connected.

The TOE is intended to be used in a physically protected environment. It is assumed that no unauthorised personnel has physical access to the TOE. Therefore all attacks to the TOE have to be performed over the network connections of the TOE. Either from the untrusted carrier network or from the connected networks and the tunnel of the untrusted traffic.

The underlying operating system and hardware are by design dedicated for the TOE and can not be used for other tasks or applications.

It is assumed that the underlying hard- and firmware operates according to their specifications and have no security critical side-effects on the operation of the TOE. The underlying hard- and firmware are not part of this TOE, but of course the functions of the TOE rely on them.

Furthermore the TOE is assumed to operate in an environment where interception of radiation is covered by other environmental measures. The evaluation will therefore not address vulnerabilities caused by emanation from the TOE.

Remote administrators and operators of the TOE authenticated with a TLS certificate, as well as local administrators, are considered to be trustworthy. The TOE will not protect itself against an administrator who tries to bring the TOE into an insecure

state. It is also assumed that administrators are well trained, reducing the risk that they accidentally make security critical administration mistakes.

3.1 Threat Environment

This section describes the threat model for the TOE and identifies the individual threats that are assumed to exist in the TOE environment.

The **assets** to be protected by the TOE are:

1. Organisational data within networks, the TOE enforces a strict communication policy that controls all data flows between networks. This prevents unauthorized traffic and protects organizational data from being exploited.
1. The TSF, the configuration data (including encryption keys) and the audit data, in particular the availability of the TSF to legitimate users, especially to authorized administrators through the designated management interfaces.

The exact definitions of TSF and TSF data is given by the actual design and implementation of the TOE.

The **threat agents** having an interest in manipulating the TOE and TSF behaviour to gain access to these assets can be categorized as:

1. Unauthorized third parties (“attackers”, such as malicious remote users, parties, or external IT entities) which are unknown to the TOE and its runtime environment, but may attempt to interact with the TOE. Attackers are traditionally located outside the organizational environment that the TOE is employed to protect, but may include organizational insiders too.
2. Authorized administrators of the TOE are assumed to be trustworthy, trained and to follow the instructions provided to them with respect to the secure configuration and operation of the systems under their responsibility. Hence, only inadvertent attempts to manipulate the safe operation of the TOE are expected from this community.

The motivation of threat agents is assumed to be commensurate with the assurance level pursued by this evaluation, i.e., the TOE intends to resist penetration by attackers with an **enhanced-basic** attack potential.

Although the administrators are assumed to be trustworthy and trained, we cannot exclude that mistakes are being made. This as well as insider attacks may not be prevented, but are to a large extent addressed by the OSPs P.ADMACC and P.AUDIT.

3.1.1 Threats countered by the TOE

The threats specified below are addressed by the TOE.

T.CHANNEL	An attacker may impersonate a remote network by forging authentication data within the IPSec VPN tunnel.
T.INISEC	For configuration settings which are not provided by an administrator, insecure default values may be set by the TOE.
T.MEDIATE	An attacker may send information through the TOE bypassing the filter or access control system.
T.MODIFY	The attempts of an external attacker to modify data transmitted between the TOE and a remote network goes undetected.

T.ADMIN An attacker may be able to perform administration or configuration of the TOE, or gain access to administration information and configuration data, such as secret keys or audit records, or may be able to modify such data.

T.SELPRO An attacker may read, modify, or destroy TOE internal data by transmitting data to the TOE via one of its network connections that causes modification or deletion of TOE internal data.

T.UPDATE Attacker may provide malicious TOE updates or old versions of the TOE software to introduce back-doors or exploitable weaknesses into the TOE.

3.2 Organizational Security Policies

The organisational security policies are specified making demands on the accountability of administrator actions:

P.ADMACC Administrators shall be accountable for the actions they conduct by generating sufficient audit records for the actions.

P.AUDIT The TOE shall be able to record all of its security relevant actions.

P.CONFIG The TOE shall support the means to configure and manage the TSFs.

P.NTPSYNC The TOE shall allow time synchronization information to flow as specified by its filtering policies, while limiting information leakage.

P.UDPDIODE The TOE shall allow UDP traffic to flow only in one direction as specified by its filtering policies.

3.3 Assumptions

This section specifies the assumptions that must be satisfied by the TOE environment.

A.AUDIT The TOE environment must be able to receive, store and protect the audit records generated by the TOE and provide the means for the audit analysis.

A.DHPARA The Diffie-Hellman parameters of the TOE and the remote host are of good quality.

A.KEYS It is assumed that private RSA keys used for remote administration and the VPN tunnel are of high quality and not disclosed.

A.NOEVIL Authorised administrators given privileges, are competent, non-hostile and follow all their guidance; however, they are capable of error.

A.NOEMA Interception of emanation of any kind is addressed by environmental controls that reduce the signal to noise ratio for an interceptor to a level that prohibits useful evaluation of the intercepted signals.

A.PHYSEC The TOE is physically secure, i.e. no unauthorised persons

have physical access to the TOE and its underlying system.

A.RELHARD The underlying hardware, firmware (BIOS and device drivers) and the operating system functions needed by the TOE to guarantee secure operation, are working correctly and have no undocumented security critical side effect on the functions of the TOE.

A.TIME The TOE environment provides the TOE with a reliable time stamp.

A.SINGEN Information cannot flow among the networks (local-local, remote-remote, and local-remote) unless it passes through the TOE.

4 Security Objectives

The security objectives provide a concise statement of the intended response to the security problem. It will describe which security needs will be addressed by the TOE and which will be addressed by the TOE environment, in the form of a statement of security objectives.

4.1 Objectives for the TOE

The following are the IT security objectives to be met by the TOE.

O.MEDIATE The TOE must mediate the flow of all information between users and IT entities on both local networks (directly connected via its interfaces) and remote networks (accessed via IPSec VPN tunnels), ensuring adherence to the defined communication policy.

O.CHANNEL The TOE must establish and maintain secure channels to remote networks. It must protect the authenticity and integrity of information transmitted to and from these networks, and enable remote entities to verify the integrity of outgoing information.

O.AUDIT The TOE must be able to provide audit evidence of security relevant events as well as for authorised use of security functions to allow an authorised administrator to read the audit trail.

O.CONFIG The TOE must provide the means for an authorized administrator to configure and manage the TOE security functions.

O.LIMEXT The TOE must restrict the means to configuration and control of the TOE to authorised administrators.

O.REMOTE The TOE must uniquely identify and authenticate the identity of all administrators and provide them with a secure communication channel before allowing remote administrators any access to the TOE.

O.SECSTA Upon initial start-up of the TOE or during configuration, the TOE shall provide well-defined initial settings for security relevant functions.

O.SELPRO The TOE must protect itself against attempts by attackers to

bypass, deactivate or tamper with TOE security functions.

O.UPDATE The TOE must only accept updates that are newer than the current running version and updates where the origin and integrity can be trusted.

O.NTPSYNC The TOE must allow time synchronization information to flow as specified by its filtering policies while limiting information leakage.

O.UDPDIODE The TOE must allow UDP traffic to flow only in one direction, as dictated by its security policy.

4.2 Objectives for the Operational Environment

The following are the security objectives for the operational environment of the TOE that are necessary for the TOE to meet its security objectives.

Thus, the following environmental objectives may partly be IT specific and partly related to administrative methods and/or procedural measures.

OE.AUDIT The TOE environment must be able to receive, store and protect the audit records generated by the TOE and provide the means for the audit analysis.

OE.DHPARA The Diffie-Hellman parameters of the TOE and the remote host are safe and not arbitrarily generated.

OE.KEYS It is assumed that private RSA keys used for remote administration and the VPN tunnel are of high quality and not disclosed.

OE.NOEVIL Authorised administrators and operators given privileges, are competent, non-hostile and follow all their guidance; however, they are capable of error.

OE.NOEMA Interception of emanation of any kind is addressed by environmental controls that reduce the signal to noise ratio for an interceptor to a level that prohibits useful evaluation of the intercepted signals.

OE.PHYSEC The TOE is physically secure, i.e. no unauthorised persons have physical access to the TOE and its underlying system.

OE.RELHARD The underlying hardware, firmware (BIOS and device drivers) and the operating system functions needed by the TOE to guarantee secure operation, are working correctly and have no undocumented security critical side effect on the functions of the TOE.

OE.TIME The TOE environment provides the TOE with a reliable time stamp.

OE.SINGEN The environment must guarantee that information cannot flow among the networks (local-local, remote-remote, and local-remote) unless it passes through the TOE.

4.3 Security Objectives Rationale

4.3.1 Security Objectives Coverage

The following tables provide a mapping of security objectives both for the TOE and

the TOE environment to the environment defined by the threats, policies and assumptions, illustrating that each security objective for the TOE covers at least one threat or policy, and that each security objective for the TOE environment covers at least one policy, threat or assumption.

	T:CHANNEL	T.INISEC	T.MEDIATE	T.MODIFY	T.ADMIN	T.SELPRO	T.UPDATE	P.ADMACC	P.AUDIT	P.CONFIG	P.NTPSYNC	P.UDPDIODE	A.AUDIT	A.DHPARA	A.KEYS	A.NOEVIL	A.NOEMA	A.PHYSEC	A.RELHARD	A.TIME	A.SINGEN
O.MEDIATE			X																		
O.CHANNEL	X			X																	
O.AUDIT								X	X												
O.CONFIG										X											
O.LIMEXT					X																
O.REMOTE					X			X													
O.SECSTA		X																			
O.SELPRO						X															
O.UPDATE							X														
O.NTPSYNC											X										
O.UDPDIODE												X									
OE.AUDIT								X	X				X								
OE.DHPARA				X										X							
OE.KEYS				X	X										X						
OE.NOEVIL																X					
OE.NOEMA																	X				
OE.PHYSEC					X													X			
OE.RELHARD																			X		
OE.TIME								X	X											X	
OE.SINGEN																					X

Table 1: Security objective coverage

4.3.2 Security Objectives Sufficiency

The following rationale provides justification that the security objectives are suitable to counter each individual threat and that each security objective tracing back to a threat actually contributes to the mitigation of that threat.

Threat	Rationale for the security objectives
T.CHANNEL	By requiring the TOE to establish secure channels for communication with remote networks via IPSec VPN tunnels, which includes mechanisms for authenticating the remote network during channel establishment, that protects transmitted information against unauthorized disclosure and detect any modification as in O.CHANNEL, the threat T.CHANNEL is mitigated. This process ensures that only traffic from authenticated remote sources is accepted, its integrity verified, and any attempts at unauthorized access or tampering are prevented.
T.INISEC	By requiring well-defined default setting in O.SECSTA, an initial insecure configuration of the TOE is prevented and the threat

Threat	Rationale for the security objectives
	T.INISEC of insecure configuration due to lack of administrator settings is removed.
T.MEDIATE	By demanding that the TOE must mediate (i.e. control) all data sent between networks connected to the TOE as in O.MEDIATE, this ensures that only data conforming to the defined policy is permitted to pass through the TOE, regardless of the direction. This prevents an attacker from bypassing the filter or access control system by sending unauthorized information through the TOE.
T.MODIFY	By requiring the TOE to be able to provide a secure channel against disclosure and against modification as in O.CHANNEL, the threats of T.MODIFY are being met. This is also supported by the assumption that the cryptographic parameters and keys provided by the environment are secure as in OE.DHPARA and OE.KEYS.
T.ADMIN	The threat of a non-administrator performing administration of the TOE as in T.ADMIN is addressed by requiring that all remote administration is performed using a remote secure channel, based on OE.KEYS, with identified and authorized administrators as in O.REMOTE, and by further restricting administration to these administrators as in O.LIMEXT. T.ADMIN is only relevant for the remote administration, because no unauthorized personal can physically access the TOE as demanded by OE.PHYSEC. OE.PHYSEC fulfils A.PHYSEC.
T.SELPRO	By protecting itself against bypass, deactivation and tampering as in O.SELPRO, the threat T.SELPRO is diminished to an acceptable level.
T.UPDATE	By only accepting newer versions of the TOE, the TOE will prevent that older versions with possibly known vulnerabilities will be installed (O.UPDATE). The TOE will also verify the origin and integrity of the TOE version and thereby prevent malicious version from being installed (O.UPDATE).

Table 2: Security objectives rationale for the threats

The following rationale provides justification that the security objectives of the TOE and the TOE environment are suitable to address each OSP and that each security objective tracing back to a OSP actually contributes in addressing the OSP.

OSP	Rationale for the security objectives
P.ADMACC	The auditing of administrator actions as in O.AUDIT, assisted by correct time delivery in OE.TIME and unique identification in O.REMOTE, satisfies the organisational security policy of administrators being accountable for their actions as in P.ADMACC. It is further supported by OE.AUDIT that the TOE environment must be able to receive, store and protect the audit records generated by the TOE and provide the means for the audit analysis.
P.AUDIT	The auditing of security relevant reactions is addressed by O.AUDIT, assisted by correct time delivery in OE.TIME, satisfies the organisational security policy that the TOE shall be able to record all of its security relevant actions as in P.AUDIT. It is further supported by OE.AUDIT that the TOE environment must be able to receive, store and protect the audit records generated by the TOE and provide the means for the audit analysis.
P.CONFIG	The management the TOE security functions is addressed by O.CONFIG, which ensures that the TOE provides the means for an

OSP	Rationale for the security objectives
	authorized administrator to configure and manage the TOE security functions as in P.CONFIG.
P.NTPSYNC	The synchronization of the time information as in P.NTPSYNC is addressed by O.NTPSYNC, which ensures that the TOE allows time synchronization information to flow as specified by its filtering policies, while limiting information leakage.
P.UDPDIODE	The restriction of UDP traffic as in P.UDPDIODE is addressed by O.UDPDIODE, which ensures that the TOE allows UDP traffic to flow in only one direction.

Table 3: Security objectives rationale for the OSPs

The following rationale provides justification that the security objectives of the TOE environment are suitable to address each individual assumption and that each security objective tracing back to an assumption actually contributes in addressing the assumption.

Assumption	Rationale for the security objectives
A.AUDIT	Addressed by OE.AUDIT which is identical to the assumption
A.DHPARA	Addressed by OE.DHPARA which is identical to the assumption
A.KEYS	Addressed by OE.KEYS which is identical to the assumption
A.NOEVIL	Addressed by OE.NOEVIL which is identical to the assumption
A.NOEMA	Addressed by OE.NOEMA which is identical to the assumption
A.PHYSEC	Addressed by OE.PHYSEC which is identical to the assumption
A.RELHARD	Addressed by OE.RELHARD which is identical to the assumption
A.TIME	Addressed by OE.TIME which is identical to the assumption
A.SINGEN	Addressed by OE.SINGEN which is identical to the assumption

Table 4: Security objectives rationale for the assumptions

5 Extended Components Definition

The extended requirement, FPT_TUD_EXT.1 for trusted updates is used to specify the SFR for automatic trusted updates. It has been based on the extended component which defined by [NDPP] Protection Profile for Network Devices published by NIAP in December 2023.

5.1 FPT_TUD_EXT – Trusted Updates

Family behaviour

Components in this family address the requirements for updating the TOE firmware and/or software. This is a new family defined for the FPT class.

Component levelling

FPT_TUD_EXT.1 is not hierarchical.

Management

While management functions have been specified as part of this component already, the following actions could be considered for the management functions in FMT: Ad-

ministrator initiation of updates, activation and deactivation of automatic updates, time for initiation of updates or specification of certificates used for signature verification.

Audit

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

1. Minimum: Software update.

2. Minimum: Failure of verification (digital signature, published hash or version number)

5.1.1 FPT_TUD_EXT.1 Trusted Update

Hierarchical to: none

Dependencies: FCS_COP.1 Cryptographic operation

FPT_TUD_EXT.1.1 The TSF shall provide administrators the ability to query the current version of the TOE software.

FPT_TUD_EXT.1.2 The TSF shall provide a mechanisms that [select: on a regular basis initiates, gives administrators the ability to initiate] updates to TOE software.

FPT_TUD_EXT.1.3 The TSF shall provide a means to verify software updates to the TOE using a [selection: digital signature mechanism, published hash] prior to installing those updates.

FPT_TUD_EXT.1.4 The TSF shall provide a means to verify software updates to the TOE to ensure that software update version is newer than the current version of the TOE prior to installing those updates.

Application note: The digital signature mechanism and hash mechanisms referenced in the third element must be specified in FCS_COP.1. The ST author should choose the mechanism implemented by the TOE; it is acceptable to implement both mechanisms.

6 Security Requirements

6.1 TOE Security Functional Requirements

The following table gives an overview of the functional components from the Common Criteria Part 2 that are relevant for this TOE.

Component	Component Name
FAU_GEN.1	Audit data generation
FCS_CKM.1	Cryptographic key generation
FCS_CKM.2	Cryptographic key distribution (iteration a) and Cryptographic key establishment (Refinement) (iteration b, c)
FCS_CKM.6	Timing and event of cryptographic key destruction
FCS_COP.1	Cryptographic operation (iteration a, b, c, d)

Component	Component Name
FDP_ACC.2	Complete access control
FDP_ACF.1	Security attribute based access control
FDP_IFC.2 (VPN)	Subset information flow control
FDP_IFC.2 (Filters)	Subset information flow control
FDP_IFF.1 (VPN)	Simple security attributes
FDP_IFF.1 (Filters)	Simple security attributes
FIA_ATD.1	User attribute definition
FIA_UAU.2	User authentication before any action
FIA_UID.2	User identification before any action
FMT_MOF.1	Management of security functions behaviour
FMT_MSA.1	Management of security attributes
FMT_MSA.3	Static attribute initialisation
FMT_MTD.1	Management of TSF data (iteration a, b)
FMT_SMR.1	Security roles
FMT_SMF.1	Specification of management functions
FPT_FLS.1	Failure with preservation of secure state
FPT_STM.1	Reliable time stamps
FPT_TST.1	Self test
FPT_TUD_EXT.1	Trusted update
FTP_ITC.1	Inter-TSF trusted channel
FTP_TRP.1	Trusted Path

Table 5: Functional Requirements on the TOE

The following paragraphs give an overview on the functional requirements listed in the table above with respect to the TOE. They serve as an introduction to the detailed definition of the functional requirements, which are presented in the next section.

Class FAU components are selected to describe the capability of the TOE to generate, read and protect audit data. The TOE generates audit data for events associated with the communication links it monitors. Administrators are able to select the audited events, and the log data is transferred, read and analysed in the environment.

Class FCS contains the requirements related to cryptographic operations. There are two areas in which the TOE performs cryptographic operations: Remote administration using TLS and the IPSec VPN connection which is also using TLS and SKUT for key management. The TOE is importing certificates for the remote administration and for the VPN.

Class FDP contains the security requirements associated with the access control between remote administrators and configuration data of the TOE. The class also contains the security requirements associated with the information flow control between the interfaces of the TOE. Information flow control is enforced both by the TOE subsystems as well as by the VPN tunnel.

Class FIA contains the security requirements for identification and authentication of a remote administrator performing administrative tasks. The authentication is relying

on the TLS authentication using X.509 certificates.

Class FMT contains the security management requirements. This includes the security management role of the administrator, the management functions available to the administrator and that the initial default values of the TOE will be well-defined.

Class FPT contains the requirement for the protection of the TSF, which contains the requirements for the preservation of secure state, reliable time stamps, self test and trusted update.

Class FTP contains requirements for trusted communication path between the TSF and other trusted IT products, i.e. the VPN connection.

6.1.1 Security Functional Policies implemented by the TOE

6.1.1.1 TRAFFIC SFP

The TOE will implement the information flow control policy SFP named TRAFFIC SFP to manage all data flows across its network interfaces. The TSF shall enforce the SFP on the traffic of both local networks (i.e., networks directly connected via the TOE's interfaces) and remote networks (i.e., networks accessed via IPSec VPN tunnels).

The TRAFFIC SFP applies to the physical interfaces of the TOE that can be configured by the administrator. It does not apply to the physical admin interface.

The policy enforces that all incoming packets are processed through the following steps:

- If a packet is directed to the TOE:
  - ICMPv4 and SKUT traffic are passed to the Control Plane.
  - ESP traffic is decrypted and the payload is passed to a filter module (FILTER SFP)
- If a packet is not for the TOE, it is passed to a filter module (FILTER SFP).

No other traffic shall be routed using the TRAFFIC SFP.

6.1.1.2 FILTER SFP

The TOE will implement the information flow control policy SFP named FILTER SFP. The TSF shall enforce the SFP on traffic designated by the administrator to traverse specific filters modules installed within the TOE, regulating information flow to and from connected networks.

The TSF shall apply the following rules to all traffic using the FILTER SFP:

- NTP-Filter: Enables NTP time synchronization traffic while minimizing information leakage. This filter permits synchronization data to flow as specified by the communication policy.
- UDP-Diode Filter: Permits UDP packets to flow in only a single direction as specified by the communication policy, ensuring that reverse traffic is blocked.

Built-in filters include the Allow filter (which passes packets unmodified) and the Deny filter (which drops packets).

This policy ensures that only traffic conforming to the administrator's configuration

is allowed between local networks and remote networks (or between segments within a local network).

After filtering:

- Packets intended for a remote network are encrypted and sent via the VPN.
- Packets intended for a local network are transmitted directly.
- Packets that do not match any policy are dropped.

6.1.1.3 ADMINISTRATOR ACCESS SFP

The TOE will implement the access control policy ADMINISTRATOR ACCESS SFP. The TSF shall enforce identification and authentication of remote administrators and operators before giving any administrative access to the TOE (i.e. giving any access to TSF data).

6.1.2 Class FAU – Security Audit

6.1.2.1 FAU_GEN.1 – Audit Data Generation

FAU_GEN.1.1 The TSF shall be able to generate audit data 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) the following events:

1. **connect and disconnect of a VPN channel (including connection attempts)**
2. **renegotiation of the VPN key (IPSec)**
3. **operations performed by remote administrators and operators (including connection attempts)**
4. **changes to the configuration (including filter related events).**

FAU_GEN.1.2 The TSF shall record within the audit data 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;

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP, PP-Module, functional package or ST: **none**

Application note: Although the TOE is capable to generate additional audit events, these events are not considered security events and not necessary to satisfy the TOE security objectives and therefore not considered part of the TSF.

6.1.3 Class FCS – Cryptographic Support

6.1.3.1 FCS_CKM.1 Cryptographic key generation (AES)

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **as defined in the TLS v1.3 standard [RFC8446] for AES-256 [FIPS197]** and specified cryptographic key sizes **256 bit (AES-256)** that meet the following: **generation and exchange of session keys as defined in the TLS v1.3 standard with the cipher suites defined in FCS_COP.1 (b)**.

Application note: The session keys are negotiated and established during an TLS session for remote administration (i.e. remote administrators and operators) as well as the VPN for symmetrical encryption and integrity protection of VPN packets. The TLS standard allows other cryptographic algorithms and key sizes, but only AES-256 is supported. This functionality is provided by the TOE, which can act as both a TLS server and client for VPN connections. For remote administration, the TOE acts solely as a TLS server, while the administration client provides corresponding functionality on the TLS client's side (as part of the environment).

6.1.3.2 FCS_CKM.2 (a) – Cryptographic Key Distribution (cert)

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **of digital certificates** that meets the following: **X.509 Version 3 [RFC5280]**.

Application note: This requirement addresses the exchange of X.509 certificates as part of the TLS authentication of the remote administration (i.e. remote administrators and operators) and the key management of the VPN channel.

6.1.3.3 FCS_CKM.2 (b) – Cryptographic Key Establishment (Refinement)

FCS_CKM.2.1 The TSF shall **perform** cryptographic key **establishment** in accordance with a specified cryptographic key **establishment** method **TLS handshake using a hybrid key exchange DH_RSA key exchange and eFrodoKEM-1344-SHAKE [FRODOKEM] of AES-256 session keys** that meets the following: **TLS v1.3 [RFC8446] and [TTLS].**

Application note: This component is a refinement of "FCS_CKM.2 - Cryptographic Key Distribution" as defined in CC:2022. In alignment with NDcPP v3.0e, the term "Cryptographic Key Establishment" is used to more precisely describe cases where session keys are derived through key exchange mechanisms. This requirement addresses the exchange of quantum-resistant AES-256 session keys as part of the TLS handshake protocol using Diffie-Hellman RSA + FrodoKEM for peer configuration used as part of the VPN. No other key exchange methods are accepted.

6.1.3.4 FCS_CKM.2 (c) – Cryptographic Key Establishment (Refinement)

FCS_CKM.2.1 The TSF shall **perform** cryptographic keys **establishment** in accordance with a specified cryptographic key **establishment** method **TLS handshake using DH_RSA key exchange of AES-256 session keys** that meets the following: **TLS v1.3**

[RFC8446].

Application note: This component is a refinement of "FCS_CKM.2 - Cryptographic Key Distribution" as defined in CC:2022. In alignment with NDcPP v3.0e, the term "Cryptographic Key Establishment" is used to more precisely describe cases where session keys are derived through key exchange mechanisms. This requirement addresses the exchange of AES-256 session keys as part of the TLS handshake protocol using Diffie-Hellman RSA for remote administration. No other key exchange methods are accepted.

6.1.3.5 FCS_CKM.6 – Timing and event of cryptographic key destruction

FCS_CKM.6.1 The TSF shall destroy **cryptographic keys** when **no longer needed**.

FCS_CKM.6.2 The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method: **overwriting the memory area** that meets the following: **none**.

6.1.3.6 FCS_COP.1 (a) – Cryptographic Operation (RSA-PSS)

FCS_COP.1.1 The TSF shall perform **digital signature generation and verification** in accordance with a specified cryptographic algorithm **RSA-PSS [RFC8017]** and cryptographic key sizes of **2048 bit** or **3072 bit** or **4096 bit** or **7680 bit** or **8192 bit** that meet the following: **[PKCS1v2.2] and SHA-256 [FIPS180-4]**.

Application note: This requirement addresses the RSA digital signature generation and verification operations using the RSA algorithm as required by the TLS session establishment protocol used in the VPN key exchange and the remote administration.

6.1.3.7 FCS_COP.1 (b) – Cryptographic Operation (AES-GCM)

FCS_COP.1.1 The TSF shall perform **encryption and decryption** in accordance with a specified cryptographic algorithm **AES-256 in GCM mode** and cryptographic key sizes **256 bit** that meet the following: **[FIPS197] and [NIST SP 800-38D]**.

Application note: This is used by TLS in the remote administration and for the IPSec payload in VPN. If a client or VPN node tries to use any other cipher, the client or peer will be rejected by the TOE. The encrypted data gets authenticated by its associated authentication tag in accordance with the GCM mode.

6.1.3.8 FCS_COP.1 (c) – Cryptographic Operation (RSA-PSS)

FCS_COP.1.1 The TSF shall perform **signature verification** in accordance with **RSA-PSS [RFC8017]** and cryptographic key sizes of **2048 bit** or **3072 bit** or **4096 bit** or **7680 bit** or **8192 bit** that meet the following: **RSA-PSS [RFC8017] and SHA-256 [FIPS180-4]**.

Application note: This requirement addresses the RSA digital signature verification operations using the RSA algorithm as required to verify TOE updates.

6.1.3.9 FCS_COP.1 (d) – Cryptographic Operation (KeyedHash)

FCS_COP.1.1 The TSF shall perform keyed-hash message authentication in

accordance with a specified cryptographic algorithm **implicit** in accordance with a specified cryptographic algorithm **AES-256 in GCM mode** that meets the following: **[FIPS197] and [NIST SP 800-38D]**.

Application note: All data encrypted with AES in GCM mode gets associated with an authentication tag in accordance with the specification. Additional authenticated data (AAD) gets incorporated into the tag in accordance to respective protocol (IPSec and TLS).

6.1.4 Class FDP – User Data Protection

6.1.4.1 FDP_ACC.2 – Complete access control

FDP_ACC.2.1 The TSF shall enforce the **ADMINISTRATOR ACCESS SFP** on **the subjects:**

- **remote administrators**
- **operators**

and objects:

- **configuration data of the TOE**
- **audit data locally stored in the TOE**

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

FDP_ACC.2.2 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.

6.1.4.2 FDP_ACF.1 – Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the **ADMINISTRATOR ACCESS SFP** to objects based on the following:

subject remote administrator and operator:

- **TLS certificate**
- **CN, UID or Email field of the certificate**

objects (configuration data of the TOE, resources in the internal network):

- **none.**

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

- **If the CN, UID or Email field of the subject's certificate is part of a list managed by the TOE that allows to connect as TLS client to the TOE, the user is allowed access to resources on the TOE.**

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

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

If the client certificate is not signed by a certificate of a certification authority trusted by the TOE, then access is

denied.

6.1.4.3 FDP_IFC.2 (VPN) – Complete information flow control

FDP_IFC.2.1 The TSF shall enforce the **TRAFFIC SFP** on **incoming data packets based on incoming interface and when using IPSec on the destination address of the packets** 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.

Application note: Under the TRAFFIC SFP, packets originating on local networks (directly connected interfaces) shall be forwarded into the IPSec VPN tunnel when destined for a remote network. Packets arriving from a remote network are accepted only if they are carried within an established VPN channel and directed toward a local destination. In an IPSec routed configuration, the TOE functions as a router and forwards packets from local networks to the appropriate remote network using administrator-supplied routing information.

6.1.4.4 FDP_IFC.2 (Filters) – Complete information flow control

FDP_IFC.2.1 The TSF shall enforce the **FILTER SFP** on **incoming and outgoing data packets** 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.

Application note: The FILTER SFP shall ensure that traffic designated for processing by specific filter modules (such as the NTP-filter and UDP-diode) is managed in accordance with administrator-defined rules. In particular, it governs NTP time synchronization traffic and UDP traffic, ensuring that information flow between networks is controlled as specified.

6.1.4.5 FDP_IFF.1 (VPN) – Simple security attributes

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

subjects

- **network interface**

information

- **IP packet (IPsec VPN)**

security attributes

- **incoming/outgoing network interface for the packet**
- **source/destination address for the packet**
- **traffic type (SKUT traffic, ICMPv4 traffic)**

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:

- **packets originating from local networks will be routed to the appropriate VPN channel, when using IPSec based on the packet's destination address**
- **packets carried within an established IPSec VPN channel will be routed to its designated local destination.**

FDP_IFF.1.3 The TSF shall enforce **no additional information flow control SFP rules.**

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:

- **data packets directed to the TOE that are not encapsulated within an established IPSec VPN channel are rejected, except for the following packets:**
  - **SKUT traffic**
  - **ICMPv4 traffic**

6.1.4.6 FDP_IFF.1 (Filters) – Simple security attributes

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

subjects

- **IT entities on connected networks that send and receive information through the TOE**

objects

- **IP packet**

security attributes

- **incoming/outgoing network interface for the packet**
- **source/destination address for the packet**
- **traffic type**

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:

- **packets from remote networks that contain NTP traffic**
- **UDP traffic is allowed only in the direction specified by the administrator's one-way rule**

FDP_IFF.1.3 The TSF shall enforce the **following information flow control SFP rules:**

- **The TOE will allow NTP time synchronization information to flow as specified by its filtering policies, while limiting the maximum possible information leakage from the trusted network to a**

maximum of 2500 bit per 5 minutes time slot.

- **The TOE will allow UDP traffic to flow only in one direction between defined networks, as specified by the filter module configuration.**

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:

- **UDP traffic flowing in the direction opposite to the one allowed.**

6.1.5 Class FIA – Identification and Authentication

6.1.5.1 FIA_ATD.1 – User Attribute Definition

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

- **identity of the remote administrator or operator in form of the CN, UID or Email record of the client certificate**
- **association of the remote administrator or operator with a TLS client certificate.**

Application note: Only remote administrators and operators are know to the TOE. Local administrators are not individually identified by the TOE, but are identified and authorized through organizational means. All certificates have to be signed by a trusted root CA. This root certificate is provided in the configuration file.

6.1.5.2 FIA_UAU.2 – User Authentication before any Action

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

Application note: Only the remote administrators and operators are subject to any authentication by the TOE. Authentication is performed by verifying that the administrator or operator possesses the private key part to the TLS client certificate. Local administrators are authenticated through organizational means, by only allowing authorized personnel physical access to the TOE.

6.1.5.3 FIA_UID.2 – User Identification before any Action

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

Application note: Identification of remote administrators and operators is performed by presenting a TLS client certificate.

6.1.6 Class FMT – Security Management

6.1.6.1 FMT_MOF.1 – Management of security functions behaviour

FMT_MOF.1.1 The TSF shall restrict the ability to **modify the behaviour of** the functions **listed below** to **an administrator**:

- **change the configuration of the TOE**
- **change the configuration of filters**

6.1.6.2 FMT_MSA.1 – Management of security attributes

FMT_MSA.1.1 The TSF shall enforce the **ADMINISTRATOR ACCESS SFP** to restrict the ability to **modify** the security attributes **consisting of possible configuration options** to **administrators**.

6.1.6.3 FMT_MSA.3 – Static attribute initialisation

FMT_MSA.3.1 The TSF shall enforce the **ADMINISTRATOR ACCESS SFP** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

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

Application note: An administrator can restrict unauthenticated access and specify security relevant initial values by changing the rules in the configuration file.

6.1.6.4 FMT_MTD.1 (a) – Management of TSF data (administrator)

FMT_MTD.1.1 The TSF shall restrict the ability to **query or modify** the **TSF data listed below** to **an administrator:**

- **audit data [query]**
- **status of the TOE [query]**
- **status of the VPN-tunnels [query]**
- **configuration files [query, modify].**

Application note: The configuration files do not include the cryptographic key and seed file on the USB storage or smart card. Status of the TOE includes the status of the interfaces, firmware versions and other information that are security relevant for the administrator.

6.1.6.5 FMT_MTD.1 (b) – Management of TSF data (operator)

FMT_MTD.1.1 The TSF shall restrict the ability to **query** the **TSF data listed below** to **an operator:**

- **audit data [query]**
- **status of the TOE [query]**
- **status of the VPN-tunnels [query]**
- **configuration files [query].**

Application note: The configuration files do not include the cryptographic key and seed file on the USB storage or smart card. Status of the TOE includes the status of the interfaces, firmware versions and other information that are security relevant for the administrator.

6.1.6.6 FMT_SMF.1 – Specification of management functions

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

- **query the software version**

- **query interface status**
- **query interface statistics**
- **query tunnel status**
- **apply changes to the configuration file**
- **reboot the TOE**
- **initiate update of the TOE software**
- **query the audit files**
- **install and manage loadable filters**

Application note: The security management functions related to changes of the configuration of the TOE are described in more detail in FMT_MOF.1.

6.1.6.7 FMT_SMR.1 – Security roles

FMT_SMR.1.1 The TSF shall maintain the roles:

- **administrator**
- **operator**

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

Application note: The administrator is verified by the TOE for remote administrator login. When the administrator is local the TOE environment ensures that only administrators have local access to the TOE (OE.PHYSEC).

6.1.7 Class FPT – Protection of the TOE Security Functions

6.1.7.1 FPT_FLS.1 – Failure with preservation of secure state

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

- **invalid configuration file**
- **removing of the USB memory or smart card**
- **failed integrity verification**

Application note: If the new configuration file is unreadable or does not conform with the syntax as described in the user guidance, then the previous configuration file will be kept in use. The syntax of the configuration file has been designed to prevent insecure configurations. In other failure cases the TOE shuts down.

6.1.7.2 FPT_STM.1 – Reliable time stamps

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

Application note: The TOE relies on its environment (i.e. the operating system) for reliable time stamps. The operating system can be configured to utilize the ntpd daemon to synchronize with another external system.

6.1.7.3 FPT_TST.1 – TSF testing

FPT_TST.1.1 The TSF shall run a suite of self tests **during initial start-up, periodically during normal operation** to demonstrate the correct operation of **the TSF**:

- **check file integrity**
- **check encryption engine**

FPT_TST.1.2 The TSF shall provide authorised users with the capability to

verify the integrity of **none**.

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

Application note: During startup the firmware image is verified, but not the configuration data. The fsintegrity daemon performs integrity check verification for all control and data plane modules before they are started and then periodically. In addition, the data plane performs a verification check of its encryption engine at startup, using reference test vectors [RFC3686], and on the start of each VPN packet.

6.1.7.4 FPT_TUD_EXT.1 Trusted Update

FPT_TUD_EXT.1.1 The TSF shall provide administrators the ability to query the current version of the TOE software.

FPT_TUD_EXT.1.2 The TSF shall provide a mechanisms that **on a regular basis initiates and gives administrators the ability to initiate** updates to TOE software.

FPT_TUD_EXT.1.3 The TSF shall provide a means to verify software updates to the TOE using a **digital signature mechanism** prior to installing those updates.

FPT_TUD_EXT.1.4 The TSF shall provide a means to verify software updates to the TOE to ensure that software update version is newer than the current version of the TOE prior to installing those updates.

Application note: The trusted update is both an automatic mechanisms as well as administrator controlled. Apart from this the TOE (and filters) can be updated outside of the control of the TOE by authorized persons that have physical access to the TOE (relying on the OE.PHYSEC). The activation and deactivation of the automatic update mechanism is part of the configuration changes made to the configuration file described in FMT_MOF.1 and FMT_SMF.1.

6.1.8 Class FTP – Trusted path/channels

6.1.8.1 FTP_ITC.1 Inter-TSF Trusted Channel (IPSec)

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

FTP_ITC.1.2 The TSF shall permit **the TSF or a remote trusted IT product** to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for **VPN services.**

Application note: This channel is the VPN communication channel (IPSec) that the TOE may establish with other remote TOEs. Note that the communication channel can be established either by the TOE or by the remote TOE (remote end of the VPN).

6.1.8.2 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 and 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 **initial user authentication and all remote management functions.**

6.2 Security Functional Requirements Rationale

6.2.1 Coverage

The following table provides a mapping of SFR to the security objectives, showing that each security functional requirement addresses at least one security objective.

SFR	Security Objectives
FAU_GEN.1	O.AUDIT
FCS_CKM.1	O.CHANNEL, O.REMOTE
FCS_CKM.2 (a)	O.CHANNEL, O.REMOTE
FCS_CKM.2 (b)	O.CHANNEL
FCS_CKM.2 (c)	O.REMOTE
FCS_CKM.6	O.SELPRO
FCS_COP.1 (a)	O.CHANNEL, O.REMOTE
FCS_COP.1 (b)	O.CHANNEL, O.REMOTE
FCS_COP.1 (c)	O.UPDATE
FCS_COP.1 (d)	O.CHANNEL, O.REMOTE
FDP_ACC.2	O.LIMEXT
FDP_ACF.1	O.LIMEXT, O.REMOTE
FDP_IFC.2 (VPN)	O.MEDIATE, O.SELPRO
FDP_IFC.2 (Filters)	O.NTPSYNC, O.UDPDIODE
FDP_IFF.1 (VPN)	O.MEDIATE, O.SELPRO
FDP_IFF.1 (Filters)	O.NTPSYNC, O.UDPDIODE
FIA_ATD.1	O.REMOTE
FIA_UAU.2	O.REMOTE, O.SELPRO
FIA_UID.2	O.REMOTE, O.SELPRO
FMT_MOF.1	O.CONFIG, O.LIMEXT, O.SELPRO
FMT_MSA.1	O.LIMEXT, O.SELPRO
FMT_MSA.3	O.CONFIG
FMT_MTD.1 (a)	O.CONFIG, O.SELPRO
FMT_MTD.1 (b)	O.CONFIG, O.SELPRO
FMT_SMF.1	O.CONFIG

SFR	Security Objectives
FMT_SMR.1	O.REMOTE
FPT_FLS.1	O.SECSTA, O.SELPRO
FPT_STM.1	O.AUDIT
FPT_TST.1	O.SECSTA, O.SELPRO
FPT_TUD_EXT.1	O.UPDATE
FTP_ITC.1	O.CHANNEL
FTP_TRP.1	O.REMOTE

Table 6: Security functional requirement coverage

6.2.2 Sufficiency

The following rationale provides justification for each security objective for the TOE, showing that the security functional requirements are suitable to meet and achieve the security objectives.

Security Objective	Rationale
O.MEDIATE	The objective: • The TOE must mediate the flow of all information between users and IT entities on both local networks and remote networks in accordance with its security policy. is met by: • FDP_IFC.2 (VPN), which enforces the TRAFFIC SFP on incoming traffic. • FDP_IFF.1 (VPN), which defines rules for the TRAFFIC SFP.
O.CHANNEL	The objective: • The TOE must be able to provide trusted channels to remote networks and protect information transmitted to and received from these networks against unauthorized disclosure. It must also detect any modification of incoming information and provide the means for remote entities to verify the integrity of information sent from the TOE. is met by: • FCS_CKM.1, which specifies the cryptographic key generation for SKUT • FCS_CKM.2 (a), which specifies the certificate handling for SKUT • FCS_CKM.2 (b), which specifies the cryptographic key establishment for SKUT • FCS_COP.1 (a), which specifies RSA-PSS for SKUT • FCS_COP.1 (b) (d), which specifies AES-256-GCM for IPSec • FTP_ITC.1, which mandates a dedicated encrypted communication channel
O.AUDIT	The objective: • The TOE must be able to provide audit evidence of security relevant events as well as for authorised use of security functions to allow an authorised administrator to read the audit trail. is met by: • FAU_GEN.1, which specifies the list of audit events • FPT_STM.1, which mandates reliable time stamps
O.CONFIG	The objective: • The TOE must provide the means for an authorized

Security Objective	Rationale
	administrator to configure and manage the TOE security functions. is met by: • FMT_MOF.1, which specifies the list of management actions • FMT_MSA.3, which allows for different initial values • FMT_MTD.1 (a), which lists the management functions for administrators • FMT_MTD.1 (b), which lists the management functions for operators • FMT_SMF.1, which lists the TSF security management functions
O.LIMEXT	The objective: • The TOE must restrict the means to configuration and control of the TOE to authorised administrators. is met by: • FDP_ACC.2, which mandates the ADMINISTRATOR ACCESS SFP • FDP_ACF.1, which specifies the identification of administrators and operators • FMT_MOF.1, which limits the management actions to administrators • FMT_MSA.1, which limits the configuration options to authorised administrators
O.REMOTE	The objective: • The TOE must uniquely identify and authenticate the identity of all administrators and provide them with a secure communication channel before allowing remote administrators any access to the TOE. is met by: • FCS_CKM.1, which specifies the cryptographic key generation for remote administration • FCS_CKM.2 (a), which specifies the certificate handling for remote administration • FCS_CKM.2 (c), which specifies the cryptographic key establishment for remote administration • FCS_COP.1 (a), which specifies RSA-PSS for remote administration • FCS_COP.1 (b) (d), which specifies AES-256-GCM for remote administration • FDP_ACF.1, which specifies the authentication for remote administration • FIA_ATD.1, which specifies the identity attributes for remote administration authentication • FIA_UAU.2, which mandates authentication for remote administration • FIA_UID.2, which requires identification for remote administration • FMT_SMR.1, which specifies the user roles known to the TSF • FTP_TRP.1, which mandates authenticated and encrypted trusted path for remote administration.
O.SECSTA	The objective: • Upon initial start-up of the TOE or during configuration, the TOE shall provide well-defined initial settings for security relevant functions. is met by: • FPT_FLS.1, which specifies how the TSF preserves a

Security Objective	Rationale
	secure state in case of a failure (e.g., incorrect configuration) • FPT_TST.1, which specifies the self tests for the TSF to ensure correct operation of the TOE
O.SELPRO	The objective: • The TOE must protect itself against attempts by attackers to bypass, deactivate or tamper with TOE security functions. is met by: • FCS_CKM.6, which specifies that cryptographic keys are deallocated after use • FDP_IFC.2 (VPN), which enforces the TRAFFIC SFP on incoming traffic. • FDP_IFF.1 (VPN), which rejects traffic from the external network outside the trusted channel • FIA_UAU.2, which requires users to be authenticated before any TSF-mediated action is allowed • FIA_UID.2, which requires users to be identified before any TSF-mediated action is allowed • FIA_MOF.1, which restricts configuration changes and restarts to administrators • FMT_MSA.1, which limits configuration changes to authorised administrators • FMT_MTD.1 (a), which limits configuration changes to authorised administrators • FMT_MTD.1 (b), which limits operators to read-only actions • FPT_FLS.1, which specifies how the TSF preserves a secure state in case of a failure • FPT_TST.1, which specifies the self tests of the TSF to ensure correct operation of the TOE.
O.UPDATE	The objective: • The TOE must only accept updates that are newer than the current running version and updates where the origin and integrity can be trusted. is met by: • FPT_TUD_EXT.1, which specifies trusted updates are verified and accepted. • FCS_COP.1 (c) provides the cryptographic operation for verification of signatures of the software image.
O.NTPSYNC	The objective: • The TOE must allow time synchronization information to flow as specified by its filtering policies, while limiting information leakage. is met by: • FDP_IFC.2 (Filters), which enforces the FILTER SFP on incoming traffic. • FDP_IFF.1 (Filters), which allows NTP time synchronization information to flow as specified by its filtering policies, while limiting the maximum possible information leakage from the trusted network.
O.UDPDIODE	The objective: • The TOE must allow UDP traffic to flow only in one direction. is met by: • FDP_IFC.2 (Filters), which enforces the FILTER SFP on incoming traffic. • FDP_IFF.1 (Filters), which allows UDP traffic to flow only in one direction, according to the configured SFP traffic flow control rules.

Table 7: Security functional requirement sufficiency

6.2.3 Security Requirements Dependency Analysis

Dependencies within the EAL package selected (EAL4) for the security assurance requirements have been considered by the authors of CC Part 3 and are not analysed here again. The augmentation by flaw remediation, ALC_FLR.1, has no dependencies on other requirements. The security functional requirements in this Security Target do not introduce dependencies on any security assurance requirement; neither do the security assurance requirements in this Security Target introduce dependencies on any security functional requirement.

The following table demonstrates the dependencies of SFRs modelled in CC Part 2 and how the SFRs for the TOE resolve those dependencies.

SFR	Dependencies	Resolution
FAU_GEN.1	FPT_STM.1	Yes, by FPT_STM.1
FCS_CKM.1	[FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1] [FCS_RBG.1 or FCS_RNG.1] FCS_CKM.6	Yes, by FCS_CKM.2 (iteration a, b, c) addressed by the operational environment (underlying system) Yes, by FCS_CKM.6
FCS_CKM.2 (a) (cert)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5]	No, but covered by A.PHYSEC and A.NOEVIL
FCS_CKM.2 (b) (key establishment)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5]	Yes, by FCS_CKM.1
FCS_CKM.2 (c) (key establishment)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5]	Yes, by FCS_CKM.1
FCS_CKM.6	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	Yes, by FCS_CKM.1
FCS_COP.1 (a) (RSA-PSS)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	No, since the keys are imported Yes, by FCS_CKM.6
FCS_COP.1 (b) (AES-GCM)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	Yes, by FCS_CKM.1 Yes, by FCS_CKM.6
FCS_COP.1 (c) (RSA-PSS)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	No, since the public-key is part of the an existing image. No, since the public-key is public.
FCS_COP.1 (d)	[FDP_ITC.1 or	Yes, by FCS_CKM.1

SFR	Dependencies	Resolution
(KeyedHash)	FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	Yes, by FCS_CKM.6
FDP_ACC.2	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.2 Yes
FDP_IFC.2 (VPN)	FDP_IFF.1	Yes, by FDP_IFF.1 (VPN)
FDP_IFC.2 (Filters)	FDP_IFF.1	Yes, by FDP_IFF.1 (Filters)
FDP_IFF.1 (VPN)	FDP_IFC.1 FMT_MSA.3	Yes, by FDP_IFC.2 (VPN) Yes
FDP_IFF.1 (Filters)	FDP_IFC.1 FMT_MSA.3	Yes, by FDP_IFC.2 (Filters) Yes
FIA_ATD.1	No dependencies	–
FIA_UAU.2	FIA_UID.1	Yes, by FIA_UID.2
FIA_UID.2	No dependencies	–
FMT_MOF.1	FMT_SMR.1 FMT_SMF.1	Yes and by A.PHYSEC and A.NOEVIL Yes
FMT_MSA.1	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	Yes, by FDP_ACC.2 Yes Yes
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	Yes Yes
FMT_MTD.1 (a) (administrator)	FMT_SMR.1 FMT_SMF.1	Yes Yes
FMT_MTD.1 (b) (operator)	FMT_SMR.1 FMT_SMF.1	Yes Yes
FMT_SMF.1	No dependencies	–
FMT_SMR.1	FIA_UID.1	Yes, by FIA_UID.2
FPT_FLS.1	No dependencies	–
FPT_STM.1	No dependencies	–
FPT_TST.1	No dependencies	–
FPT_TUD_EXT.1	FCS_COP.1	Yes, by FCS_COP.1 (c)
FTP_ITC.1	No dependencies	–
FTP_TRP.1	No dependencies	–

Table 8: Security functional requirements dependency analysis

6.3 Security Assurance Requirements

The assurance requirements are the EAL4 package augmented with ALC_FLR.1.

Assurance class	Assurance components
ADV: Development	ADV_ARC.1 – Security architecture description
	ADV_FSP.4 – Complete functional specification
	ADV_IMP.1 – Implementation representation of the TSF

	ADV_TDS.3 – Modular design
AGD: Guidance documents	AGD_OPE.1 – Operational user guidance
	AGD_PRE.1 – Preparative procedures
ALC: Life-cycle support	ALC_CMC.4 – Production support, acceptance procedures and automation
	ALC_CMS.4 – Problem tracking CM coverage
	ALC_DEL.1 – Delivery procedures
	ALC_DVS.1 – Identification of security measures
	ALC_LCD.1 – Developer defined life-cycle model
	ALC_TAT.1 – Well defined developer tools
	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.2 – Analysis of coverage
	ATE_DPT.1 – Testing: basic design
	ATE_FUN.1 – Functional testing
	ATE_IND.2 – Independent testing – sample
AVA: Vulnerability assessment	AVA_VAN.3 – Focused vulnerability analysis

Table 9: Security assurance requirements

6.3.1 Security Assurance Requirements Rationale

The assurance level EAL4 augmented with ALC_FLR.1 has been chosen as appropriate for a network device that separates directly connected (local) networks from remote networks accessed via IPSec VPN tunnels, since it provides a moderate to high level of independently assured security, and a thorough investigation of the TOE. Since it can be expected that the mechanisms separating a sensitive internal network from remote, unclassified networks will be subject to enhanced-basic attack potential, choosing EAL4 makes the TOE capable of meeting this attack potential.

By choosing EAL4, a description of the basic modular design of the TOE and a subset of the implementation is required. This will support the model of separating the base platform and management functionality from the VPN separation mechanisms. This will also support the model of adding additional separating mechanisms such as proxies or filters without affecting the VPN functionality or the security of the underlying platform.

To complement the assurance provided by EAL4, the ALC_FLR.1 augmentation addresses the need for ongoing flaw remediation. This capability is essential in high-stakes environments where new vulnerabilities may be discovered post-deployment. ALC_FLR.1 ensures that any identified flaws can be systematically resolved, maintaining the TOE's security over its operational lifecycle. Given the importance of securing assets across different security classifications, the ALC_FLR.1 component aligns with both regulatory requirements and organizational security expectations by enabling controlled updates and security maintenance throughout the device's

lifespan.

It is assumed that the TOE is operated in an environment where attackers have average expertise of the involved systems (e.g., general and publicly available knowledge on network protocols), limited resources and may have an average motivation because of possible high-value assets protected by the TOE. The overall attack potential is assumed to be enhanced-basic, which means that EAL4 is considered an appropriate assurance level, because it contains AVA_VAN.3 which ensures resistance against attackers with enhanced-basic attack potential. This level of assurance is needed in environments where the confidence in the ability of the TOE to provide a high degree of separation is necessary such as the separation of environments with different security classifications.

7 TOE Summary Specification

The TOE summary specification identifies the security functions that the TOE implements to meet the requirements defined in chapter 6 to the security target.

7.1 SF.PKTCLASS – Packet classification

Incoming traffic to the TOE is initially classified by the Flow Multiplexor, which determines whether the traffic originates from a local network (i.e. directly connected interfaces) or from a remote network (i.e. via an IPSec VPN tunnel).

The classification follows these rules:

- Packets directed to the TOE:
  - ICMPv4 and SKUT Traffic: Passed directly to the Control Plane.
  - ESP Traffic: Decrypted by the “Packet Encryption” component; its payload is then handed off to the FILTER SFP for further processing.
- Packets not directed to the TOE: All such packets are passed to a filter module, where the FILTER SFP is applied. If no matching filter is found, the packet is dropped.

Note: This classification supports all traffic flows (local-to-local, local-to-remote, remote-to-local, and remote-to-remote) by ensuring that only packets conforming to the defined policies are permitted.

This behaviour maps to the following SFRs:

- For VPN traffic: FDP_IFC.2 (VPN) and FDP_IFF.1 (VPN)
- For non-VPN (filter) traffic: FDP_IFC.2 (Filters) and FDP_IFF.1 (Filters)

7.2 SF.FILTER – Filter Functionality

The TOE provides a filter functionality that allows the enforcement of communication policies between different networks. The filters operate according to the FILTER SFP and control the flow of the information based on administrator-defined rules. All filters are installed by the administrator. In the evaluated configuration, two filters are available: NTP Filter and UDP-diode Filter.

Built-in filters include the Allow filter (which passes packets unmodified) and the Deny filter (which drops packets).

After have been processed by the filter module:

- Packets intended for a remote network are encrypted and sent via the IPSec VPN tunnel.
- Packets intended for a local network are transmitted directly.
- Packets not matching any policy are dropped.

It maps to the following SFRs:

- FMT_SMF.1

7.3 SF.VPN – VPN Functionality

The VPN functionality has two parts, key negotiation and bulk encryption.

Key negotiation is done by the skut component (fskutd) using the SKUT protocol described in [SKUT5]. The skut component will generate encryption keys according to the TLS v1.3 protocol (FCS_CKM.1). The TLS_AES_256_GCM_SHA384 is the only supported TLS cipher suite. The TLS certificate and keys used in the key negotiation are RSA based (FCS_COP.1 (a), FCS_CKM.2 (a) and FCS_CKM.2 (b)). The TLS certificate and private key is either in a file or in a smart card device. The SKUT protocol itself uses AES-256-GCM (FCS_COP.1 (b) and FCS_COP.1 (d)).

Bulk encryption is done in the “Packet Encryption” component. The encryption uses IPSec [RFC4303] with AES-256-GCM (FCS_COP.1 (b) and FCS_COP.1 (d)).

This section also maps to the FTP_ITC.1 SFR.

7.4 SF.AUDIT – Security Audit

The central part of the security audit system is the syslog server component (Tlogd). It accepts audit records from other components over local sockets and sends the records to a remote log server over a TCP-connection. If the log server isn't available the log records are kept in RAM.

The syslog server component generates audit records for start-up and shutdown of the audit system as well as adding date and time of the record.

The following audit records are generated:

1. Connect and disconnect of a VPN-channel
2. Renegotiation of the VPN key
3. Operations performed by remote administrators and operators
4. Changes to the configuration and keys

It maps to the FAU_GEN.1 SFR.

7.5 Security Management

The TOE offers administrators several interfaces to configure and manage the TSF. This includes the front panel, the USB interface and remote administration using the HTTPS interface.

The control panel and USB interface assume physical access to the TOE and are available to authorised administrators. Identification and authentication of administrators is only performed for remote administrators.

7.5.1 SF.REMADM – Remote administration

Remote administration is handled by the remadm component (httpd). All remote administrators and operators are authenticated using a TLS v1.3 encrypted and mutually authenticated connection to the remadm component. (FCS_CKM.1, FCS_CKM.2 (a) and FCS_CKM.2 (c)). The TLS_AES_256_GCM_SHA384 is the only supported TLS cipher suite (FCS_COP.1 (b)). The local TOE TLS certificate/ private key is stored in a file on USB memory or on a smart card (FCS_CKM.2 (a) and FCS_CKM.2 (c)).

Users are authorised as either an administrator or an operator role. Each role has a list of CN, UID or Email addresses that when it matches a field in the client certificate will authorise the user for that role.

It is the remadm component that decides which actions are allowed for each role. The operator role can read status of the device, audit files and the configuration. The administrator role can also change the configuration.

The remadm component sends commands to the mgmtd and statusd components to handle configuration and status control. Audit files are read directly by the remadm component.

This section maps also to the following SFRs:

- FDP_ACC.2
- FDP_ACF.1
- FIA_ATD.1
- FIA_UAU.2
- FIA_UID.2
- FMT_MSA.1
- FMT_MTD.1 (a)
- FMT_MTD.1 (b)
- FMT_SMR.1
- FTP_TRP.1

7.5.2 SF.MGMT

Status and statistics of the TOE are managed through the statusd component. It interacts with other components to provide its services.

This section maps the following SFRs:

- FMT_MOF.1
- FMT_SMF.1

7.5.3 SF.CONF – Configuration

The TOE is configured through a configuration file in text format. It can be edited outside of the TOE by exporting it on a USB-memory or changed using the remote administration.

The syntax of the configuration file is described in the user guidance.

This section maps to the following SFRs:

- FMT_MSA.3
- FMT_SMF.1

7.6 TSF protection and support functions

7.6.1 SF.CRYPTO – Cryptographic Support

All cryptographic functions in the TOE is implemented using Tapir, an encryption library developed by Tutus.

This maps to the following SFRs:

- FCS_CKM.1
- FCS_CKM.2 (a)
- FCS_CKM.2 (b)
- FCS_CKM.2 (c)
- FCS_COP.1 (a)
- FCS_COP.1 (b)
- FCS_COP.1 (c)
- FCS_COP:1 (d)

7.6.2 SF.KEYDEST – Timing and event of cryptographic key destruction

Memory segments that are used for cryptographic keys are overwritten as soon as they are no longer needed.

This section maps to FCS_CKM.6

7.6.3 SF.CONFVER – Configuration verification

The TOE verifies the configuration file during import. If the file is unreadable or contains syntactical errors, then it will be rejected and the previous configuration, if any, stays in place.

This section maps to the FPT_FLS.1.

7.6.4 SF.SELFTEST – Self-test

Self tests are run both at startup and at regular intervals.

After tlogd, fsintegrity is the first component that is started after the kernel is bootstrapped. It checks the integrity of all system files before it starts any other services. Once the system is running fsintegrity will perform similar integrity checks of the system files at regular intervals.

The data plane component validates that all interfaces are available and performs a series of encryption tests. The encryption tests, which are based on test vectors from RFC 3686, verify the correct cryptographic operations of “packet encryption” component.

A test validating proper working of the “packet encryption” component is performed on per packet basis as well. For each packet undergoing cryptographic transformation, a part of the packet is encrypted and verified with a reference implementation of

crypto algorithm.

This functionality implements the following SFRs:

- FPT_FLS.1
- FPT_TST.1

7.6.5 SF.FAILSAFE – Failsafe

When the USB-memory or smart card holding the TLS certificate/private key is removed the mgmtd component is signalled, which in turn restarts the system.

This functionality implements FTP_FLS.1.

7.6.6 SF.TIME – Time-stamps

The TOE provides reliable time stamps for its own use, in particular for the generation of audit records and validation of certificates used for VPN and administrator authentication. The time stamp is provided by the TOE environment through the NTP service of the underlying operating system.

This functionality implements FPT_STM.1.

7.6.7 SF.AUTOUPD – Automatic update

The automatic update functionality is implemented using digitally singed updates that are verified using a factory installed certificate. The update system will verify the signature and verify that the software in the update is newer than the running version. This functionality applies to both the system and installed filters, allowing the administrator to update TOE components seamlessly.

This functionality implements the following SFRs:

- FPT_TUD_EXT.1
- FCS_COP.1 (c)

8 Abbreviations, Terminology and References

8.1 Abbreviations

AES	Advanced Encryption Standard
API	Application Programming Interface
CA	Certificate Authority
CC	Common Criteria
CN	Common Name
CPU	Central Processing Unit
CRL	Certificate Revocation List
DHCP	Dynamic Host Configuration Protocol
EAL	Evaluation Assurance Level
ESP	Encapsulating Security Payload

FIPS	Federal Information Processing Standard
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IPSec	Internet Protocol Security
NTP	Network Time Protocol
OSI	Open Systems Interconnection
PKCS	Public-Key Cryptographic Standard
RAM	Random Access Memory
RFC	Request For Comment
RSA	Rivest, Shamir and Adleman
SHA	Secure Hash Algorithm
SFP	Security Function Policy
SKUT	Simple Key-exchange Using TLS
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions
USB	Universal Serial Bus
VPN	Virtual Private Network

8.2 Terminology

8.3 References

CC	Information Technology – Security Techniques – Evaluation Criteria for IT Security, also known as the Common Criteria or CC – Common Criteria for Information Technology Security Evaluation. • Part 1: Introduction and general model, November 2022, Revision 1, CCMB-2022-11-001 • Part 2: Security functional Components, November 2022, Revision 1, CCMB-2022-11-002 • Part 3: Security Assurance Components, November 2022, Revision 1, CCMB-2022-11-003 • Part 4: Framework for the specification of evaluation methods and activities, November 2022, Revision 1, CCMB-2022-11-004 • Part 5: Pre-defined packages of security requirements, November 2022, Revision 1, CCMB-2022-11-005
CEM	Common Methodology for Information Technology Security Evaluation, CEM:2022, Evaluation Methodology, November 2022, Revision 1, CCMB-2022-11-006
CCERR	Errata and Interpretation for CC:2022 (Release 1) and CEM:2022

	(Release 1), Version 1.1, CCMB-2024-07-002
FIPS197	Advanced Encryption Standard (AES), 26. November 2001, FIPS-197
FIPS180-4	FIPS 180-4, Secure Hash Standard (SHS), 2012 March, FIPS PUB 180-4
FIPS198-1	FIPS 198-1, The Keyed-Hash Message Authentication Code (HMAC), 2008 March, FIPS PUB 198-1
NIST SP 800-38D	NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, 2007 November, NIST SP 800-38D
PKCS1v2.2	PKCS #1 v2.2: RSA Cryptography Standard, RSA Laboratories, October 2012
RFC3686	RFC 3686: Using Advanced Encryption Standard (AES) Counter Mode With IPSec Encapsulation Security Payload (ESP)
RFC4303	RFC 4303: IP Encapsulating Security Payload (ESP), The Internet Society, December 2005
RFC5280	RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, NIST, May 2008
RFC8017	RFC 8017: PKCS #1: RSA Cryptography Specifications Version 2.2, Internet Engineering Task Force (IETF), November 2016
RFC8446	RFC 8446: The Transport Layer Security (TLS) Protocol Version 1.3
SKUT5	SKUT v5, revision 0.2, 2023-06-01
FRODOKEM	FrodoKEM: Learning With Errors Key Encapsulation, 2024-12-05
TTLS	TLS 1.3 in TTLS, revision 1.6, 2021-06-22