Datum
2011-11-07

FMV Dokumentbeteckning

Utgåva
2.0

Ansv område/Enhet
AK Gem, Gen Prodstöd

Klassificeringsnr

Sida
1 (59)

Er referens

Ert datum

Er beteckning

FMV tjänsteställe, handläggare, telefon
FMV, Thomas Dahlbeck, 08-782 52 15

FMV föreg. datum

FMV föreg. beteckning

Protection Profile – Information Gateway

FMV
Försvarets materielverk
Box 1002
732 26 Arboga

Tel: 08-782 40 00
Fax: 0589-825 42

registrator@fmv.se
www.fmv.se

Org.nr: 202100-0340
VAT nr: SE202100-0340-01

Table of contents

1 PP INTRODUCTION 5

1.1 PP REFERENCE 5
1.2 TOE OVERVIEW 5
1.2.1 System overview 5
1.2.2 TOE components 6
1.2.3 Filtering 7
1.2.4 TOE configuration 8
1.2.5 Major security functions 8
1.2.6 Roles and Users 9
1.2.7 States 10
1.2.8 Available non-TOE hardware/software/firmware 11
1.2.9 TOE usage 11
1.2.10 Document overview 12

2 CONFORMANCE CLAIMS 13

2.1 CC CONFORMANCE CLAIM 13
2.2 PP AND PACKAGE CONFORMANCE CLAIMS 13

3 SECURITY PROBLEM DEFINITION 14

3.1 INTRODUCTION 14
3.2 THREATS 14
3.2.1 Threat agents 14
3.2.2 Assets 15
3.2.3 Threats addressed by the TOE 16
3.3 ORGANISATIONAL SECURITY POLICIES 17
3.4 ASSUMPTIONS 18

4 SECURITY OBJECTIVES 19

4.1 INTRODUCTION 19
4.2 SECURITY OBJECTIVES FOR THE TOE 20
4.3 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT 23
4.4 SECURITY OBJECTIVES RATIONALE 24
4.4.1 Security Objective Coverage 24
4.5 SECURITY OBJECTIVES SUFFICIENCY 25
4.5.1 Threats 25
4.5.1.1 T.BAD_CONFIG 25
4.5.1.2 T.ILLEGAL_CONFIG 25
4.5.1.3 T.POLICY_VIOLATION 25
4.5.1.4 T.MALFUNCTION 25
4.5.1.5 T.PHYSICAL_ATTACK 25
4.5.1.6 T.TAMPER 26
4.5.2 Organisational Security Policies 27
4.5.3 Assumptions 27

5 EXTENDED COMPONENTS DEFINITION 28

6 SECURITY REQUIREMENTS 29

6.1 SECURITY FUNCTIONAL REQUIREMENTS 29
6.1.1 Security Functional Policies 30
6.1.1.1 Audit_Data Information Flow Control Policy 30
6.1.1.2 Traffic_Data Information Flow Control Policy 30
6.1.1.3 Configuration Access Control Policy 30
6.1.2 Security Audit, FAU 30
6.1.2.1 Security alarms, FAU_ARP.1 30

6.1.2.2 Audit data generation, FAU_GEN.1 30
6.1.2.3 Potential violation analysis, FAU_SAA.1 31
6.1.2.4 Selective audit, FAU_SEL.1 32
6.1.2.5 Protected audit trail storage, FAU_STG.1 32
6.1.3 Communication, FCO 33
6.1.3.1 Selective proof of origin, FCO_NRO.1 33
6.1.3.2 Enforced proof of origin, FCO_NRO.2 33
6.1.3.3 Cryptographic operation, FCS_COP.1a (Authentication) 33
6.1.3.4 Cryptographic operation, FCS_COP.1b (Security_Configuration) 34
6.1.3.5 Cryptographic operation, FCS_COP.1c (Traffic_Data) 34
6.1.3.6 Cryptographic operation, FCS_COP.1d (Revocation_Data) 34
6.1.4 User Data Protection, FDP 35
6.1.4.1 Subset access control, FDP_ACC.1 35
6.1.4.2 Security attribute based access control, FDP_ACF.1 35
6.1.4.3 Data authentication with identity of guarantor, FDP_DAU.2 37
6.1.4.4 Subset information flow control, FDP_IFC.1a (Audit_Data) 37
6.1.4.5 Subset information flow control, FDP_IFC.1b (Traffic_Data) 37
6.1.4.6 Simple security attributes, FDP_IFF.1a (Audit_Data) 37
6.1.4.7 Simple security attributes, FDP_IFF.1b (Traffic_Data) 38
6.1.4.8 No illicit information flows, FDP_IFF.5 39
6.1.4.9 Import of user data with security attributes, FDP_ITC.2 39
6.1.5 Identification and authentication, FIA 40
6.1.5.1 User attribute definition, FIA_ATD.1 40
6.1.5.2 Timing of authentication, FIA_UAU.1 40
6.1.5.3 User identification before any action, FIA_UID.2 41
6.1.6 Security management, FMT 41
6.1.6.1 Management of security functions behaviour, FMT_MOF.1 41
6.1.6.2 Secure security attributes, FMT_MSA.2 41
6.1.6.3 Management of TSF data, FMT_MTD.1a (Dynamic_Parameters) 42
6.1.6.4 Management of TSF data, FMT_MTD.1b (TOE_Status) 42
6.1.6.5 Management of TSF data, FMT_MTD.1c (Security_Configuration) 42
6.1.6.6 Revocation, FMT_REV.1 42
6.1.6.7 Specification of management functions, FMT_SMF.1 43
6.1.6.8 Security roles, FMT_SMR.1 43
6.1.7 Protection of the TSF, FPT 44
6.1.7.1 Failure with preservation of secure state, FPT_FLS.1 44
6.1.7.2 Passive detection of physical attack, FPT_PHP.1 (Tamper_seal) 44
6.1.7.3 Notification of physical attack, FPT_PHP.2 (Lid_opening) 44
6.1.7.4 Reliable time stamps, FPT_STM.1 45
6.1.7.5 Inter-TSF basic TSF data consistency, FPT_TDC.1 45
6.1.7.6 TSF testing, FPT_TST.1 46
6.2 SECURITY ASSURANCE REQUIREMENTS 47
6.3 SECURITY REQUIREMENTS RATIONALE 48
6.3.1 Security Functional Requirements Dependencies 48
6.3.2 Security Assurance Requirements Dependencies 51
6.3.3 Security Functional Requirements Coverage 51
6.3.4 Security Functional Requirements Sufficiency 53
6.3.5 Justification of the Chosen Evaluation Assurance Level 57
APPENDIX A – ABBREVIATIONS AND ACRONYMS 58

Document history

Version	Issue date	Edited by	Revision description
0.1	2010-01-29	Annika Öberg	Basic layout and fundamentals.
0.2-1.1	2010-06-04	Tommy Ebbesson	Content
1.2-1.3	2010-10-13	Tommy Ebbesson	Updates after review. Draft version.
1.4	2011-07-05	Anders Staaf	Updated after updated requirement specifications. Early draft version not fully quality ensured. Parts of section 4.5 are TBD.
1.5	2011-07-13	Anders Staaf	Updated after early comments from Staffan Persson, Atsec. Added content to section 4.5.
1.6	2011-07-24	Anders Staaf	Updated after internal review.
1.7	2011-08-29	Anders Staaf	Updated after - TOR Application review, CSEC - ”Uppdaterad synpunkts text”, FMV, mail 2011-08-15 - ETR, Atsec, 2011-08-24, and - “Förhandskommentarer på PP” Olle Segerdahl MUST, mail 2011-08-24
1.8	2011-09-15	Anders Staaf	Updated after ETR, Atsec, v1.1, 2011-09-07.
1.9	2011-10-18	Anders Staaf	Updated after SFR errors and inconsistencies, Atsec, v1.0, 2011-10-11 and CSEC Statements, CSEC, 2011-10-14.
1.91	2011-10-21	Anders Staaf	Updated after SFR errors and inconsistencies, Atsec, v1.0 (?), 2011-10-20.
2.0	2011-11-07	Anders Staaf	Updated according to CSEC Technical Oversight Report – Garm 2, Issue 1.0, 2011-11-02.

1 PP Introduction

1.1 PP Reference

Title	Protection Profile Information Gateway
Document version	2.0
Issue date	2011-11-07
PP editor	Combitech AB, Anders Staaf

1.2 TOE Overview

The TOE is a content filtering information gateway appliance consisting of both hard- and software. The purpose of the TOE is to enable a data transfer in a secure manner between two separated networks in a way upholding the security policy concerning data export and import between the individual networks. The security policy is below referred to as the network separation security policy.

1.2.1 System overview

The TOE constitutes a common platform for medium assurance control of data transfers between two networks. One possible application of the TOE is to enable transfer of data packages, messages or files, with different sensitivity between two networks.

The TOE has four external interfaces:

- Two network interfaces,
- One administration interface to an Administration Node, and
- One log system interface to an external system for audit trail management.

All external interfaces are Ethernet (IEEE 802.3) compatible. The network interfaces communicate with TCP/IP or UDP/IP (IPv4/IPv6).

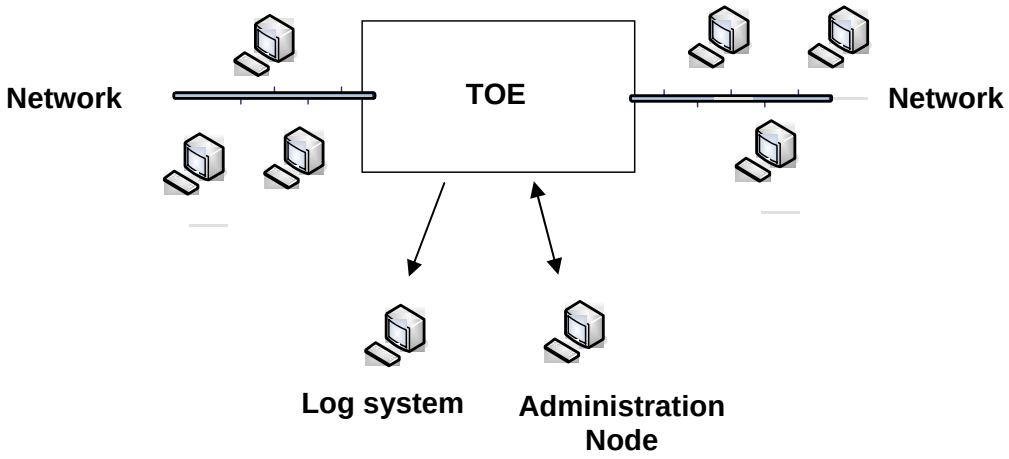

Figure 1, TOE system overview

Data received at the network interfaces are filtered according to the filter rules.

The TOE administrator uses the external Administration Node over the administration interface for identification and authentication and to perform TOE administration.

The TOE is sending audit data records over the log system interface to an external log system for storage and management of the audit trail.

1.2.2 TOE components

The TOE is divided into three separated nodes, two Service Nodes interfacing one network each and a Filter Node interfacing the Service Nodes. User data to be transferred from one network to the other has to pass all three nodes.

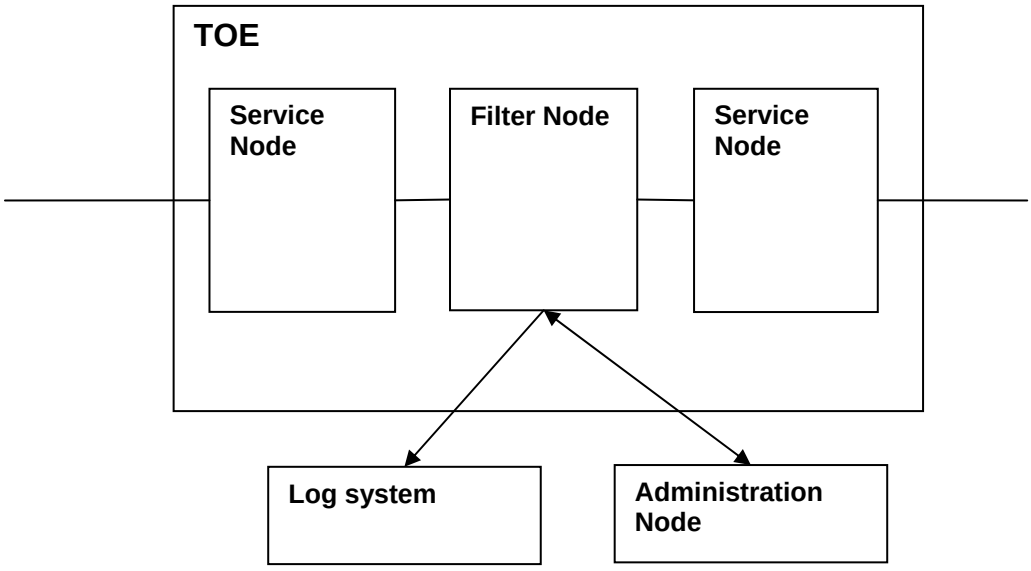

Figure 2, TOE components

The Service Nodes are able to receive and format data from the connected networks to an intermediary format that is suitable for control and filtering in the filtering node.

The nodes run upon their own dedicated hardware, CPU´s, memory, etc. A display is mounted locally on the TOE as to provide visual information about:

- TOE version and an identification of the unique TOE individual,
- The configuration,
- IP- and MAC addresses of service nodes ,
- Relevant time, e.g. internal time, time for boot and when the latest configuration was injected, and
- TOE errors

1.2.3 Filtering

The TOE offers general filter functionality, configurable by a Turing complete language. Filtering is the process of applying the filter program (written in the filter language) on a message received from a Service Node and interpreted from an intermediate format. Validation of e.g. data content, sizes, elements, attributes, etc. are possible.

It is also possible to perform integrity, authentication and non-repudiation verification of received data using digital signatures supplied by authorised senders.

The mechanisms associated to the filtering process can be outlined as to consist of a structural verification of the incoming messages or files. The steps of this process are as follows:

1. The receiving Service Node converts the incoming data to an intermediate format, e.g. XML, and sends it to the Filter Node.
2. The Filter Node verifies the syntax according to configurable syntax validation scheme.
3. The Filter Node converts the data to a configurable object model.
4. The Filter Node verifies the object model against configurable filtering rules expressed in a Turin complete filtering language.
5. If configured for integrity, authentication and non-repudiation verification, the attached digital signature is verified against the authorised sender’s public keys and certificate.
6. If the verifications do not fail the Filter Node pass on the data to the sending Service Node.
7. The sending Service Node re-converts the data to its original format and sends it to the receiving network.

If any step fails, the TOE immediately rejects the message and creates an audit record of the event.

The filter program may change the message content before re-sending, e.g. meta-data attached to the received payload data may be stripped of, preventing risks of illicit data flows through the TOE.

All security related events and a configurable amount of traffic-related events are generating audit data sent to an external log system.

1.2.4 TOE configuration

At delivery, the TOE factory setting provides initial secure settings for configuration and parameters. At first start-up after delivery it is possible load an initial configuration including what is needed to identify and authenticate at least one administrator.

The TOE functionality can be configured by the use of signed configuration files. It is possible to update and configure:

- Filter rules,
- Definitions for the intermediate format, syntax validation, and the object model,
- TOE software,
- Certificates,
- Trusted CA for user data and administration,
- Audit events and contents, and
- Other configuration items.

A trusted entity (a Configurator) creates security configuration files and appends a digital signature. The TOE Administrator injects the security configuration files into the TOE from the Administration Node. The TOE, after verifying signatures, updates the TOE configuration from the security configuration files.

The TOE administrator may set the time, modify dynamic parameters, such as IP network addresses, etc, and also request TOE status information directly from the Administration Node. The TOE Administrator has to be identified and authenticated before accessing any of the TOE administrative services.

Configuration and update of the TOE is only possible after a physical switch has been turned into configuration position. When configuration is performed the TOE physically disconnect the network interfaces. A physical switch may also be present for resetting the TOE into a factory setting stage. This optional switch shall be situated inside the TOE only reachable by breaking a tamper detection seal.

1.2.5 Major security functions

- Intrusion prevention and leakage
  The TOE enforces by its filtering capabilities the network separation security policy preventing unauthorised data between the separated networks.
- Auditing
  Auditing security and configurable traffic related events, the audit records are exported to an

external log management system. No local audit trails are to be held by the TOE. If temporarily stored within the TOE they are integrity protected.

The audit level is configurable and might depend on the characteristics of the environment in which TOE is to be used.

- Self test
  The TOE runs a suite of self tests during initial start-up, periodically during normal operation etc. to demonstrate the correct operation of the TOE security functionality.
- Secure state
  If severe errors occur, if self test of the TOE and its configuration fail, or if any predefined security related events are raised, TOE raises an alarm event and enters a blocked mode with user data transfer inhibition.
- Secure configuration
  The TOE can be securely reconfigured by the use of an administrator interface and signed files or by the use of dynamic parameters. Only authenticated Administrators are allowed to inject configuration files, modify dynamic parameters, and query TOE status. Only trusted entities (Configurators) are allowed to create and sign security configuration files. The authenticity of Administrators and Configurators is verified by the use of certificates.
- Tamper detection
  Tamper seals are located on the TOE for visual inspection. An audit event is generated if the lid is opened during operation.
- Internal TOE protection
  Data transferred between different parts of the TOE is integrity protected by physical means. The TOE and its configuration are residing in write protected memory while stored in the TOE.
- Domain separation
  The Filter and Service Nodes run in their own dedicated hardware (e.g. their own context of CPU, primary- and secondary memories)
- Verification of digital signatures
  The TOE supports the Swedish Defense PKI system. Certificates are used to identify and authenticate TOE Administrators. Digital signatures are used to protect configuration files and user data from integrity and authentication errors.

1.2.6 Roles and Users

A role is a predefined set of rules, establishing the allowed interactions between a user (person or IT entity) and the TOE. The users (persons or IT entities) get their role authorities after authenticating against certificates stored in the TOE. The following roles are defined.

- Configurator
  A person or IT entity with authority to influence the TOE security functionality by defining syntax scheme, filter rules, import certificates, configure auditable events and content, etc.

The Configurator creates and signs security configuration files to be injected by the Administrator.

- Administrator
  A person with authority to inject security configuration files, modify dynamic parameters, and query TOE status information.

The TOE recognises one non-administrative authorised user:

- Authorised_Sender
  A person or IT entity authorised to digitally sign and send user data through the TOE network interfaces, if a digital signature is requested by the filter rules.

1.2.7 States

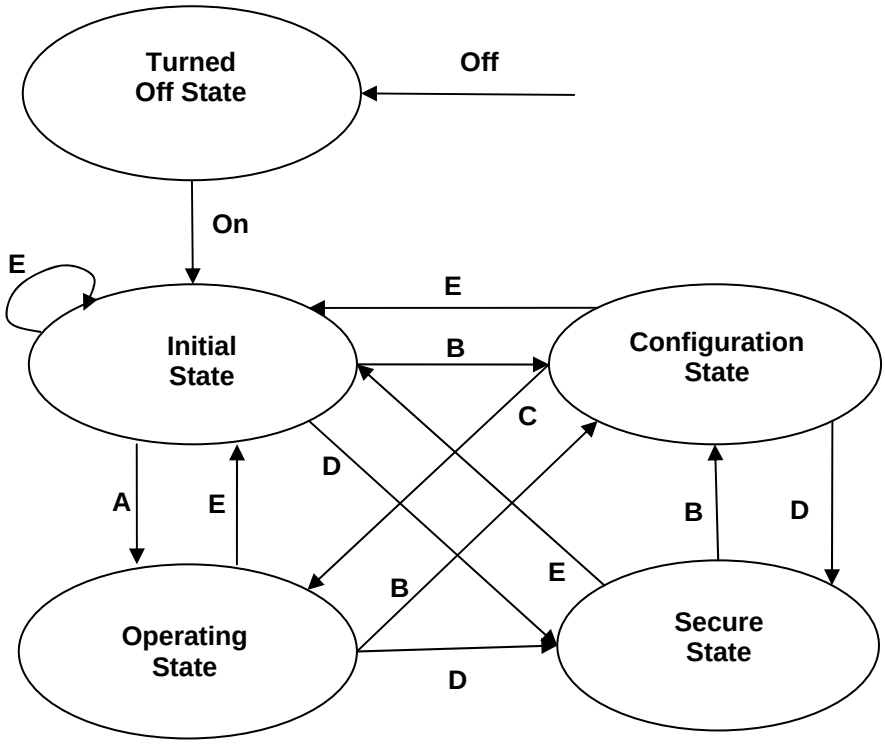

Figure 3, TOE states and state transitions

State	Description
Turned Off State	TOE is shut down, TSF is disabled, no interfaces are available.
Initial State	The TOE is performing self tests and loads the available security configuration and dynamic parameters. The network interfaces are disabled, the log system and the administration interfaces are available.
Configuration State	The Administrator may inject security configuration files, dynamic parameters and query TOE status. The network interfaces are disabled, the log system and the administration interfaces are available.
Operating State	The TOE is operating according to its configuration. All interfaces are available.
Secure State	An error has been detected. The network interfaces are disabled, the log system and the administration interfaces are available.

Table 1, TOE states

State transition	Current state	Next State	Condition
On	Turned Off State	Initial State	The TOE is turned on
Off	All states	Turned Off State	The TOE is turned off, any cleanup performed
A	Initial State	Operating State	Self test did not fail, consistent configuration loaded
B	All states	Configuration State	Configuration switch is on
C	Configuration State	Operating State	Configuration switch is off, consistent configuration loaded
D	All states	Secure State	Detected error
E (optional)	All states	Initial State	Factory reset switch is turned on

Table 2, TOE state transitions

1.2.8 Available non-TOE hardware/software/firmware

The TOE is a stand-alone device consisting of hard- and software. The external entities, log system and Administration Node, are required as to provide support for reception of audit trails from the TOE and to supply an administration interface to the TOE. Neither the log system nor the Administration Node is part of the TOE.

1.2.9 TOE usage

The opportunity for theft or tampering of the TOE shall be diminished during operation, storage or transport by physical protection. The Administration node and the channel from the Administration node shall be protected by physically or other means.

The TOE environment shall analyse and react on audit records received from the TOE and inspect the TOE and its tamper seal for signs of manipulation.

The Authorised_Senders and Administrators shall be security screened to be non-hostile and willing to follow their instructions (if human persons), before they are authorised to interact with the TOE. The Administrators shall also be sufficiently trained prior to being authorised to administrate the TOE.

1.2.10 Document overview

Chapter 1 gives a description of the PP and the TOE. This description serves as an aid to understand the security requirements and the security functions.

Chapter 2 states the conformance claims made.

In chapter 3, the security problem definition of the TOE is described. This includes assumptions about the environment of the TOE, threats against the TOE, TOE environment and organizational security policies that are to be employed to ensure the security of the TOE.

The Security Objectives stated in chapter 4 describes the intent of the Security Functions. The Security Objectives are divided into two groups of security objects, for the TOE and for the TOE environment.

No extended components are defined which leaves chapter 5 empty.

In chapter 6 the IT security functional and assurance requirements are stated for the TOE. These requirements are a selected subset of the requirements of part 2 and 3 of the Common Criteria standard.

2 Conformance Claims

2.1 CC Conformance Claim

This Protection Profile is Common Criteria Part 2 conformant and Common Criteria Part 3 conformant. Conformance is claimed to the following versions:

Common Criteria for Information Technology Security Evaluation

- Part 2: Security Functional Components, July 2009, Version 3.1, Revision 3, CCMB-2009-07-002
- Part 3: Security Assurance Components, July 2009, Version 3.1, Revision 3, CCMB-2009-07-003

The Protection Profile follows the structure given in Common Criteria Part 1, Introduction and general model, July 2009, Version 3.1, Revision 3, CCMB-2009-07-001 using the guidance from ISO/IEC TR 15446:2009(E) Information technology - Security techniques - Guide for the production of protection profiles and security targets, second edition.

2.2 PP and Package conformance Claims

This Protection Profile does not claim conformance to any other Protection Profile.

This Protection Profile claims conformance to assurance requirement package EAL 4 augmented by ALC_FLR.1.

This PP requires demonstrable PP conformance.

3 Security Problem Definition

3.1 Introduction

The purpose of the security problem definition, SPD, is to define the scope and nature of the security problem the TOE is intended to address.

The environment to which the TOE shall cope with is defined as a number of assets, threat agents, threats, assumptions and policies.

The SPD consists of identified assumptions about the environment, threats to assets and organisational security policies.

3.2 Threats

3.2.1 Threat agents

Agent	Description
Logical_Attacker	A person or IT entity that has no authorities in the TOE but logical access to the TOE by its interfaces. The Logical_Attacker has malicious intents, is highly motivated, and rich in resources and time.
Physical_Attacker	A person that has no authorities in the TOE but has gained temporary physical access to the TOE in operation, storage or during transport. The Physical_Attacker has malicious intents, is highly motivated, rich in resources but lacks opportunity.
Administrator	A person with administrative authorities in the TOE. The Administrator has deep knowledge of the management and the functionalities of the TOE and is well trained but still capable of making errors. The Administrator has no malicious intent and any threats associated with this agent are unintentional.
Sender	A person or IT entity using the TOE as a black box, sending data through the TOE network interfaces without malicious intent. The Sender has no deeper knowledge about the TOE and any threats associated with this agent are unintentional.

Table 3, Threat agents

3.2.2 Assets

Asset	Description	Type of data
Traffic_Data	Data in transit through the TOE from one network to the other. The data may be in the form of messages or files.	User
User_Resources	Network and software resources at rest within the TOE environment and accessible through the TOE interfaces.	User
Security_Configuration	Files for TOE security functionality configuration loaded in the TOE. Includes always: - Security_Configuration file version number, - hardware identification for the TOE units addressed for the security functionality configuration, and - version numbers for all TOE Software parts. May also include: - Definition of the intermediate format used between the Filter and Service Nodes, - scheme definitions for Traffic_Data syntax validation, - definition of object model used for validation of the filter rules, - filter rules for Traffic_Data validation written in a Turing complete language, - audit events and data configuration, - CA, Configurator, Administrator, and Authorised_Sender certificates, - TOE_TSF software updates, and/or - other configuration items.	TSF
Dynamic_Parameters	Dynamically changeable configuration parameters loaded in the TOE. The following non-exhaustive list of parameters shall be dynamically changeable: - IP addresses for external network interfaces, - Netmask, - Default Gateway, - DNS settings, and - MTU (Maximum Transmission Unit).	TSF
TOE_TSF	Software and hardware implementing TOE security functions.	TSF
Revocation_Data	CRL and OCSP revocation data.	TSF
Audit_Data	Audit data generated by the TOE.	User / TSF

Table 4, Assets

3.2.3 Threats addressed by the TOE

Threat	Description	Assets affected						
		Traffic_Data	User_Resources	Security_Configuration	Dynamic_Parameters	TOE_TSF	Revocation_Data	Audit_Data
T.BAD_CONFIG	Administrator manipulates or circumvents by mistake the TOE security functionality by introducing malicious code or corrupting the configuration through the administration interface.	X	X	X	X	X	X	X
T.ILLEGAL_CONFIG	Logical_Attacker attempts to - modify or destroy authorised configuration files, - inject unauthorised configuration files, - modify or destroy dynamic parameters, or - inject malicious code into the TOE by unauthorised access through the administration interface.	X	X	X	X	X	X	X
T.POLICY_VIOLATION	Logical_Attacker or Sender, intentionally or unintentionally, attempts to send unauthorised data through the TOE network interfaces, violating the network separation security policy.		X					
T.MALFUNCTION	An error occurs, causing TOE to enter an uncontrolled state where asset compromise or breakage of the network separation security policy may occur.	X	X	X	X	X	X	X
T.PHYSICAL_ATTACK	Physical_Attacker attempts to physically tamper the TOE to manipulate or circumvent the TOE security functionality.	X	X	X	X	X	X	X
T.TAMPER	Logical_Attacker or Sender, intentionally or unintentionally, attempts to manipulate or circumvent the TOE security functionality or TSF data by introducing malicious code into the TOE through the network interfaces.	X	X	X	X	X	X	X

Table 5, Threats addressed by the TOE

3.3 Organisational Security Policies

OSP	Description
P.AUDIT	The following events shall be audited: - All security related events, - all configuration changes, - integrity verification results concerning TOE_TSF, Security_Configuration, and Dynamic_Parameters, - all detected hardware and software errors, including self test verification failures, - detected critical temperature levels of the TOE, and - if the TOE lid is opened during Operating State. It shall also be possible to log the following events and reports: - All traffic and filter related events, - configurable amount of payload data and other information from Traffic_Data, - traffic related status reports, and - heartbeats The following attributes shall be logged for Traffic_Data, if applicable: - File name or information about message, - file or message size, - checksum (hash) computed over the file or message, - file or message type, - application protocol, - originator of the file or message, - recipient of the file or message, and - if the file or message is accepted or otherwise the reason for blocking. Date and time and the TOE unit's hardware identification shall be included in all audit records. All configuration changes, i.e. injection of Security_Configuration and modification of Dynamic_Parameters shall be traceable to an Administrator.
P.AUDIT_TRANSFER	The audit records from the Filter and the Service Nodes shall be transferred from the Filter Node to an external log system. Audit records shall be availability and integrity protected if temporary stored within the TOE.
P.DOMAIN_SEPARATION	The Filter and Service Nodes shall run in their own dedicated hardware (e.g. their own context of CPU, primary- and secondary memories).
P.PKI	The TOE shall support the Swedish Defense PKI system. The TOE shall support certificates of the type X.509 v3. It shall be possible to import more than one trusted CA certificates. It shall be possible to revoke certificates listed in CRL or in OCSP revocation data published by a trusted CA. It shall be possible to import revocation data manually and automatically with configurable intervals into the TOE.
P.QUERY_TOE_STATUS	Administrators shall be able to query: - the TOE unit hardware identification, - the version number of TOE software parts, - the version number of Security_Configuration files, - the values of Dynamic_Parameters, and - other TOE status values.

Table 6, Organisational security policies

3.4 Assumptions

Assumption	Description
A.AUDIT_ANALYSIS	The TOE environment is assumed to analyse and react on audit records received from the TOE.
A.AUDIT_HANDLING	The TOE environment is assumed to integrity and availability protect audit records when in transit from the TOE and when stored in an external log system.
A.REVOCATION_DATA	Certification Revocation Lists, CRLs, and revocation data from OCSP, are assumed to be delivered from a trusted CA with enough periodicity to uphold trusted means for enhanced identification.
A.NOEVIL	The Authorised_Senders and Administrators are assumed to be security screened to be non-hostile and, if human persons, willing to follow their instructions, before authorised to interact with the TOE.
A.PHYSICAL_PROTECTION	A physical protection aiming to reduce the opportunity for theft or tampering during operation, storage or transport of the TOE is assumed.
A.SECURE_ADMIN_CHANNEL	The Administration node and the channel from the Administration node are assumed to be protected by physically or other means.
A.TAMPER_INSPECTION	The tamper seal and other sign of manipulation are assumed to be inspected by the Administrator before the TOE is taken into operation and regularly during operation. If improper manipulation is observed the Administrator is assumed to take the TOE out of operation.
A.TRAINING	The Administrators are assumed to be sufficiently trained prior to being authorised to administrate the TOE.

Table 7, Assumptions

4 Security Objectives

4.1 Introduction

The statement of security objectives defines the security objectives for the TOE and its environment. The security objectives intend to address all security environment aspects identified. The security objectives reflect the stated intent and are suitable to counter all identified threats and cover all identified organisational security policies and assumptions. The following categories of objectives are identified:

- The security objectives for the TOE shall be clearly stated and traced back to aspects of identified threats to be countered by the TOE and/or organisational security policies to be met by the TOE.
- The security objectives for the environment shall be clearly stated and traced back to aspects of identified threats countered by the TOE environment, organisational security policies or assumptions.

4.2 Security Objectives for the TOE

Security Objective	Description
O.AUDIT	The following events shall be audited: - All security related events, - all configuration changes, - integrity verification results concerning TOE_TSF, Security_Configuration, and Dynamic_Parameters, - all detected hardware and software errors, including self test verification failures, - detected critical temperature levels of the TOE, and - if the TOE lid is opened during Operating State. The following events and reports shall in the Security_Configuration be configurable to audit: - All traffic and filter related events, - configurable amount of payload data and other information from Traffic_Data, - traffic related status reports, and - heartbeats For Traffic_Data, at least the following attributes shall be logged, if applicable: - File name or information about message, - file or message size, - checksum (hash) computed over the file or message, - file or message type, - application protocol, - originator of the file or message, - recipient of the file or message, and - if the file or message is accepted or otherwise the reason for blocking. Date and time and the TOE unit's hardware identification shall be included in all audit records. All configuration changes, i.e. injection of Security_Configuration and modification of Dynamic_Parameters shall be traceable to an Administrator.
O.AUDIT_TRANSFER	The audit records from the Filter and the Service Nodes shall be transferred from the Filter Node to an external log system. Audit records shall be availability and integrity protected if temporary stored within the TOE.
O.DOMAIN_SEPARATION	The Filter and Service Nodes shall run in their own dedicated hardware (e.g. their own context of CPU, primary- and secondary memories).
O.ERROR_DETECTION	The TOE shall be able to detect errors in soft- and hardware and critical high temperatures. In case of a detected error the TOE shall show an error message on a display and enter Secure State.

Security Objective	Description
O.PKI	The TOE shall support the Swedish Defense PKI system. The TOE shall support certificates of the type X.509 v3. It shall be possible to import more than one trusted CA certificates. It shall be possible to define a CA as trusted publisher of Authorised_Sender's and Administrator's certificates. It shall be possible to revoke certificates listed in CRL or in OCSP revocation data published by a trusted CA. It shall be possible to import revocation data manually and automatically with configurable intervals into the TOE.
O.QUERY_TOE_STATUS	Administrators, and only Administrators, shall be able to query: - the TOE unit hardware identification, - the version number of TOE software parts, - the version number of Security_Configuration files, - the values of Dynamic_Parameters, and - other TOE status values.
O.ROBUST_TOE_ACCESS	The TOE shall provide mechanisms that control an Administrator's logical access to the TOE administration interface and to explicitly deny access to non-authorised users. The TOE shall provide mechanisms to authenticate Authorised_Senders and Configurators. Authentication of users shall be based on certificates. The users' certificate shall be verified against a CA certificate. If multiple CA certificates are used a CA certificates may be dedicated for a role or user. It shall be possible to import user certificates and information required for authentication of those users, through Security_Configuration files.

Security Objective	Description
O.SECURE_CONFIGURATION	The TOE security functionality configuration shall be updated automatically by the TOE Filter Node from Security_Configuration files. Dynamic_Parameters shall be modified from an Administration Node over the administration interface. No other way of modifying the TOE security functionality configuration or dynamically changeable parameters shall exist. At first start-up after delivery initial secure values, when applicable, of Security_Configuration and Dynamic_Parameters shall be provided as factory setting. It shall be possible to load initial Security_Configuration files including Administrator's certificate and information required for authentication at the Administration Node. The CA that has signed the initial Security_Configuration files shall be accepted as trusted. The possibility to return to the factory settings may be provided by a hardware factory reset switch available when the tamper seal is broken and the lid is opened. Injection of Security_Configuration files and modification of Dynamic_Parameters shall only be allowed in Configuration State through the administration interface. The only way to enter Configuration State shall be through a hardware switch. In Configuration State all the network interfaces shall be physically disconnected. Only authenticated Administrators shall be authorised to inject Security_Configuration files and modify Dynamic_Parameters. Prior to accepting the Security_Configuration files the following verifications shall be done: - The integrity and authenticity of the Security_Configuration files shall be verified by the means of a digital signature supplied by the Configurator. The digital signature shall be verified against a CA certificate approved for security functionality configuration updates. - The version numbers of the Security_Configuration files shall be verified to be higher than the currently loaded. - A hardware identifications provided in the Security_Configuration files shall be verified against an unique hardware identification, digitally stored in the TOE. If the signature, version number, or hardware identification verification fails, the update shall be rejected.
O.SELF_TEST	The TOE shall during start-up and periodically during operation verify the integrity of TOE_TSF, Security_Configuration, and Dynamic_Parameters. In case of failing verification the TOE shall show an error message on a display and enter Secure State.
O.TAMPER_DETECTION	The TOE shall be equipped with - tamper seals and - detection when the lid is opened during Operating State. The audit event generated on detection of the lid is opened during Operating State provides the TOE environment the possibility to detect and react on tampering attempts.
O.TOE_PROTECTION	The TOE_TSF, Security_Configuration and Dynamic_Parameters shall reside in persistent memory only writable in Configuration State and physically write protected in all other states.

Security Objective	Description
O.TRAFFIC	All data entering the TOE in the form of messages or files on the network interfaces, shall be validated against the network separation security policy which shall include - Definition of a well-defined data format used for intra-TOE communication between the Filter and the Service nodes, - data syntax rules, - definition of the object model used for verification of the data filer rules, and - data filter rules specifying specific rules for filtering data, using a Turing complete filter language. Validation of e.g. data content, sizes, elements, attributes, and digital signatures, may be performed. The data filter language shall include PKI primitives for the possibility to configure integrity, authentication and non-repudiation verification of data digitally signed by the Authorised_Sender. It shall only be allowed so pass data through the TOE network interfaces in Operational State. If the TOE is in Operational State and the data entering a TOE network interface comply with the network separation security policy, then the payload of the data shall be passed through, with or without meta data according to the applied filter rules, to the receiving network.

Table 8, Security objectives

4.3 Security Objectives for the Operational Environment

Security Objective	Description
OE.AUDIT_ANALYSIS	The TOE environment shall analyse and react on audit records received from the TOE.
OE.AUDIT_HANDLING	The TOE environment shall protect audit records when in transport from the TOE and when stored in an external log system.
OE.REVOCATION_DATA	Certification Revocation Lists, CRLs, and revocation data from OCSP, shall be delivered from a trusted CA with enough periodicity to uphold trusted means for enhanced identification.
OE.NOEVIL	The Authorised_Senders and Administrators shall be security screened or tested to be non-hostile and, if human persons, willing to follow their instructions, before authorised to interact with the TOE.
OE.PHYSICAL_PROTECTION	The TOE shall be physically protected to avoid theft or tampering during operation, storage or during transport
OE. SECURE_ADMIN_CHANNEL	The Administration node and the channel from the Administration node to the TOE administration interface shall be physically protected.
OE.TAMPER_INSPECTION	The tamper seal and other sign of manipulation shall regularly be inspected by the Administrator. If improper manipulation is observed the Administrator shall take the TOE out of operation.
OE.TRAINING	The Administrators shall be sufficiently trained prior to being authorised to administrate the TOE.

Table 9, Security objectives for the operational environment

4.4 Security Objectives Rationale

4.4.1 Security Objective Coverage

This section provides tracings between objectives for the TOE and what threats are being countered by the objective(s) and what OSPs being enforced by the security objectives.

Also the tracing between each security objective for the operational environment and the threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective is shown.

	T.BAD_CONFIG	T.ILLEGAL_CONFIG	T.POLICY_VIOLATION	T.MALFUNCTION	T.PHYSICAL_ATTACK	T.TAMPER	P.AUDIT	P.AUDIT_TRANSFER	P.DOMAIN_SEPARATION	P.PKI	P.QUERY_TOE_STATUS	A.AUDIT_HANDLING	A.AUDIT_ANALYSIS	A.REVOCATION_DATA	A.NOEVIL	A.PHYSICAL_PROTECTION	A.SECURE_ADMIN_CHANNEL	A.TAMPER_INSPECTION	A.TRAINING
O.AUDIT							X												
O.AUDIT_TRANSFER						X		X											
O.DOMAIN_SEPARATION									X										
O.ERROR_DETECTION				X		X													
O.PKI										X									
O.QUERY_TOE_STATUS											X								
O.ROBUST_TOE_ACCESS		X																	
O.SECURE_CONFIGURATION	X	X																	
O.SELF_TEST						X													
O.TAMPER_DETECTION					X														
O.TOE_PROTECTION		X				X													
O.TRAFFIC			X																
OE.AUDIT_HANDLING												X							
OE.AUDIT_ANALYSIS													X						
OE.REVOCATION_DATA														X					
OE.NOEVIL															X				
OE.PHYSICAL_PROTECTION																X			
OE.SECURE_ADMIN_CHANNEL		X															X		
OE.TAMPER_INSPECTION					X													X	
OE.TRAINING																			X

Table 10, Security objective coverage

4.5 Security Objectives Sufficiency

4.5.1 Threats

4.5.1.1 T.BAD_CONFIG

O.SECURE_CONFIGURATION states that security configuration only can be done by Security_Configuration files that are digitally signed by the Configurator, have a higher version number than the currently loaded, and are dedicated to the unique TOE individual diminishing the risk for unintentional corruption of the Security_Configuration by the Administrator. Together this diminishes the risk for unintentional corruption of the configuration.

4.5.1.2 T.ILLEGAL_CONFIG

O.ROBUST_TOE_ACCESS states that only authorised Administrators can access the administration interface. **OE.SECURE_ADMIN_CHANNEL** ensures that administration of the TSF through the administration interface is done over a physically protected channel.

O.SECURE_CONFIGURATION states that security configuration only can be done by Security_Configuration files that are digitally signed by the Configurator mitigating the risk that an attacker introduces malicious code or a configuration that is in conflict with the network separation security policy.

O.TOE_PROTECTION ensures that TOE_TSF, Security_Configuration and Dynamic_Parameters are writing protected while stored in the TOE preventing introduction of malicious code or a non-authorised configuration.

4.5.1.3 T.POLICY_VIOLATION

O.TRAFFIC ensures that data that enters the TOE network interfaces is validated against the network separation security policy as defined by the security configuration and only be allowed to be sent through if complying to the network separation security policy.

4.5.1.4 T.MALFUNCTION

O.ERROR_DETECTION states that the TOE shall be able to detect errors in soft- and hardware and critical high temperatures. In case of a detected error the TOE shall at least send an audit event and enter Secure State thus mitigating the TOE malfunction threat. **O.ERROR_DETECTION** removes the threat.

4.5.1.5 T.PHYSICAL_ATTACK

O.TAMPER_DETECTION and **OE.TAMPER_INSPECTION** states that the TOE shall be equipped with tamper seals which shall be periodically inspected. They also states that mechanisms

shall be applied to detect if the lid is opened during Operating State generating an audit event as to provide the Administrator with means to detect tamper attempts and possibly prevent an ongoing attack or diminishing the effects of an already performed physical attack. Since the assumption **A.PHYSICAL_PROTECTION** leaves only a limited opportunity for a physical attack, the objectives **O.TAMPER_DETECTION** and **OE.TAMPER_INSPECTION** mitigates the threat.

4.5.1.6 T.TAMPER

O.AUDIT_TRANSFER states that audit records shall be availability and integrity protected if temporary stored within the TOE thus preventing tampering of audit data.

O.TOE_PROTECTION ensures that TOE_TSF, Security_Configuration and Dynamic_Parameters are writing protected while stored in the TOE preventing tampering of code or configuration.

O.ERROR_DETECTION states that the TOE shall be able to detect errors in soft- and hardware. **O.SELF_TEST** ensures the integrity of the TOE_TSF, Security_Configuration and Dynamic_Parameters by verification at least at start-up and periodically during operation (the ST author is free to specify also other occasions where self test is to be performed). Together **O.ERROR_DETECTION** and **O.SELF_TEST** mitigates the risk of a corrupted TSF or configuration.

4.5.2 Organisational Security Policies

P.AUDIT is directly covered by **O.AUDIT.**

P.AUDIT_TRANSFER is directly covered by **O.AUDIT_TRANSFER.**

P.DOMAIN_SEPARATION is directly covered by **O.DOMAIN_SEPARATION.**

P.PKI is directly covered by **O.PKI.**

P.QUERY_TOE_STATUS is directly covered by **O.QUERY_TOE_STATUS.**

4.5.3 Assumptions

A.AUDIT_ANALYSIS is directly covered by **OE.AUDIT_ANALYSIS.**

A.AUDIT_HANDLING is directly covered by **OE.AUDIT_HANDLING.**

A.REVOCATION_DATA is directly covered by **OE. REVOCATION_DATA .**

A.NOEVIL is directly covered by **OE.NOEVIL.**

A.PHYSICAL_PROTECTION is directly covered by **OE.PHYSICAL_PROTECTION.**

A.SECURE_ADMIN_CHANNEL is directly covered by **OE.SECURE_ADMIN_CHANNEL**

A.TAMPER_INSPECTION is directly covered by **OE.TAMPER_INSPECTION.**

A.TRAINING is directly covered by **OE.TRAINING.**

5 Extended Components Definition

No extended components are used in the creation of this Protection Profile.

6 Security Requirements

6.1 Security Functional Requirements

All the used SFR components are listed in the table below. **Note**: All assignments etc. not explicitly closed by the PP-author must be defined by the ST-author.

Functional class	Functional family name	Component
FAU – Security Audit	Security audit automatic response (FAU_ARP)	FAU_ARP.1
	Security audit data generation (FAU_GEN)	FAU_GEN.1
	Security audit event selection (FAU_SEL)	FAU_SEL.1
	Security audit event storage (FAU_STG)	FAU_STG.1
	Security audit analysis (FAU_SAA)	FAU_SAA.1
FCO – Communication	Non-repudiation of origin (FCO_NRO)	FCO_NRO.1 FCO_NRO.2
FCS – Cryptographic Support	Cryptographic operation (FCS_COP)	FCS_COP.1
FDP – User Data Protection	Subset access control (FDP_ACC)	FDP_ACC.1
	Access control functions (FDP_ACF)	FDP_ACF.1
	Data authentication (FDP_DAU)	FDP_DAU.2
	Subset information flow control (FDP_IFC)	FDP_IFC.1
	Information flow control functions (FDP_IFF)	FDP_IFF.1 FDP_IFF.5
	Import from outside of the TOE (FDP_ITC)	FDP_ITC.2
FIA – Identification and Authentication	User attribute definition (FIA_ATD)	FIA_ATD.1
	User authentication (FIA_UAU)	FIA_UAU.1
	User identification (FIA_UID)	FIA_UID.2
FMT – Security Management	Management of functions in TSF (FMT_MOF)	FMT_MOF.1
	Management of security attributes (FMT_MSA)	FMT_MSA.2
	Management of TSF data (FMT_MTD)	FMT_MTD.1
	Revocation (FMT_REV)	FMT_REV.1
	Specification of Management Functions (FMT_SMF)	FMT_SMF.1
	Security management roles (FMT_SMR)	FMT_SMR.1
FPT – Protection of the TSF	Fail secure (FPT_FLS)	FPT_FLS.1
	TSF physical protection (FPT_PHP)	FPT_PHP.1 FPT_PHP.2
	Time stamps (FPT_STM)	FPT_STM.1
	Inter-TSF TSF data consistency (FPT_TDC)	FPT_TDC.1
	TSF self test (FPT_TST)	FPT_TST.1

Table 11, Security functional requirements

6.1.1 Security Functional Policies

The following data access- and information flow policies are being used:

6.1.1.1 Audit_Data Information Flow Control Policy

This information flow control policy regulates how generated audit data shall flow and be handled internally within the TOE and how it shall be exported from the TOE. . The SFP is defined by FDP_IFC.1a and FDP_IFF.1a.

6.1.1.2 Traffic_Data Information Flow Control Policy

The Traffic_Data information flow control policy regulates how the TSF shall sustain the network separation security policy. The SFP is defined by FDP_IFC.1b and FDP_IFF.1b. The Traffic_Data information flow control policy is referenced in FDP_IFF.5 ensuring that no illicit information flows exist to circumvent the policy. FCO_NRO.1 ensures that the TSF is able to generate evidence of original for Traffic_Data at the request of the Configurator.

6.1.1.3 Configuration Access Control Policy

The Configuration access control policy regulates the access to Security_Configuration and Dynamic_Parameters. The SFP is defined by FDP_ACC.1 and FDP_ACF.1. The Configuration access control policy is referenced in FDP_ITC.2 which ensures secure import of Security_Configuration files.

6.1.2 Security Audit, FAU

6.1.2.1 Security alarms, FAU_ARP.1

FAU_ARP.1.1 The TSF shall **show an error message on a display and [assignment: list of actions]** upon detection of a potential security violation.

Dependencies: FAU_SAA.1 Potential violation analysis

Application note: The ST author may specify other actions.

6.1.2.2 Audit data generation, FAU_GEN.1

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

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

b) All auditable events for the **detailed** level of audit; and

c) **The auditable events according to the following non-exhaustive list:**

- **All security related events,**
- **all configuration changes,**
- **integrity verification results concerning TOE_TSF, Security_Configuration, and Dynamic_Parameters,**

- **all detected hardware and software errors, including self test verification failures,**
- **detected critical temperature levels of the TOE, and**
- **if the TOE lid is opened during Operating State.**

The following events and reports shall in the Security_Configuration be configurable to audit:
- **All traffic and filter related events,**
- **configurable amount of payload data and other information from Traffic_Data,**
- **traffic related status reports, and**
- **heartbeats.**

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

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, **the TOE unit's hardware identification shall always be recorded and, for Traffic_Data, also the following non-exhaustive list of attributes, if applicable:**
- **File name or information about message,**
- **file or message size,**
- **checksum (hash) computed over the file or message,**
- **file or message type,**
- **application protocol,**
- **sender of the file or message,**
- **receiver of the file or message, and**
- **reason for blocking.**

Dependencies: FPT_STM.1 Reliable time stamps

Application note: The ST author may specify additional audit events and attributes. By "subject identity" is meant the TOE software part generating the event, e.g. self-test software, filter program, etc. By "sender" and "receiver" is meant the network addresses of the originator and recipient of the file or message.

6.1.2.3 Potential violation analysis, FAU_SAA.1

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

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

a) Accumulation or combination of

- **self test verification failure,**
- **detection of hard- or software errors, and**
- **detection of critical high temperatures**

known to indicate a potential security violation;

b) [assignment: any other rules].

Dependencies: FAU_GEN.1 Audit data generation

Application note: The ST author may specify other rules. Events indicating potential security violations, e.g. from the Service nodes, shall cause the actions specified in FAU_ARP.1.

6.1.2.4 Selective audit, FAU_SEL.1

FAU_SEL.1.1 The TSF shall be able to select the set of events to be audited from the set of all auditable events based on the following attributes:

a) **event type**
b) [assignment: list of additional attributes that audit selectivity is based upon]

Dependencies: FAU_GEN.1 Audit data generation
FMT_MTD.1 Management of TSF data

Application note: The ST author may specify additional attributes for received messages to base the selection on.

6.1.2.5 Protected audit trail storage, FAU_STG.1

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

FAU_STG.1.2 The TSF shall be able to **prevent any** modifications to the stored audit records in the audit trail.

Dependencies: FAU_GEN.1 Audit data generation

Application note: The audit trail is referring to temporarily storage of audit records within the TOE.

6.1.3 Communication, FCO

6.1.3.1 Selective proof of origin, FCO_NRO.1

FCO_NRO.1.1 The TSF shall be able to generate evidence of origin for transmitted **Traffic_Data** at the request of the **Configurator**.

FCO_NRO.1.2 The TSF shall be able to relate the **public key from the certificate** of the originator of the information, and the **digital signature** of the information to which the evidence applies.

FCO_NRO.1.3 The TSF shall provide a capability to verify the evidence of origin of information to **itself** given **no time limit.**

Dependencies: FIA_UID.1 Timing of identification

Application note: Requirements for validation of a digital signature, applied to Traffic_Data, is configurable through the Security_Configuration files by the Configurator. The validation is to be performed by the TSF itself and may be performed at any time but may only be successful during the originator certificate's validity period.

6.1.3.2 Enforced proof of origin, FCO_NRO.2

FCO_NRO.2.1 The TSF shall be able to generate evidence of origin for transmitted **Revocation_Data** at all times.

FCO_NRO.2.2 The TSF shall be able to relate the **public key from the certificate** of the originator of the information, and the **digital signature** of the information to which the evidence applies.

FCO_NRO.2.3 The TSF shall provide a capability to verify the evidence of origin of information to **itself** given **no time limit**.

Dependencies: FIA_UID.1 Timing of identification

Application note: The validation is to be performed by the TSF itself and may be performed at any time but may only be successful during the originator certificate's validity period.

6.1.3.3 Cryptographic operation, FCS_COP.1a (Authentication)

FCS_COP.1.1a The TSF shall perform **verification of digital signatures** in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithm] and

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

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

Application note: Only algorithms, key sizes, and applicable standards approved by the Swedish NCSA shall be used. Verification of digital signatures is used for authentication of Administrators.

6.1.3.4 Cryptographic operation, FCS_COP.1b (Security_Configuration)

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

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

Application note: Only algorithms, key sizes, and applicable standards approved by the Swedish NCSA shall be used. Verification of digital signatures is used for integrity and authenticity verification of Security_Configuration files.

6.1.3.5 Cryptographic operation, FCS_COP.1c (Traffic_Data)

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

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

Application note: Only algorithms, key sizes, and applicable standards approved by the Swedish NCSA shall be used. Verification of digital signatures is used for integrity and authenticity verification of Traffic_Data.

6.1.3.6 Cryptographic operation, FCS_COP.1d (Revocation_Data)

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

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

Application note: Only algorithms, key sizes, and applicable standards approved by the Swedish NCSA shall be used. Verification of digital signatures is used for integrity and authenticity verification of Revocation_Data.

6.1.4 User Data Protection, FDP

6.1.4.1 Subset access control, FDP_ACC.1

FDP_ACC.1.1 The TSF shall enforce the **Configuration access control policy** on **Administrators injecting Security_Configuration files and modifying and querying Dynamic_Parameters**.

Dependencies: FDP_ACF.1– Security attribute based access control

6.1.4.2 Security attribute based access control, FDP_ACF.1

FDP_ACF.1.1 The TSF shall enforce the **Configuration access control policy** to objects based on the following:

Subject and attribute:
Administrator – User ID, current TOE state.

Objects and attributes:
Security_Configuration – Digital signature,
version number,
hardware id,
initial values (factory setting), and
administration interface.
Dynamic_Parameters – Initial values (factory setting), and
administration interface.

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

a) Only Administrators shall be authorised to inject Security_Configuration files and to modify and query Dynamic_Parameters.

b) Prior to accepting the Security_Configuration files the following verifications shall be done:

- **The integrity and authenticity of the Security_Configuration files shall be verified by the means of a digital signature supplied by the**

Configurator. The digital signature shall be verified against a CA certificate trusted for security functionality configuration updates.

- **The version numbers of the Security_Configuration files shall be verified to be higher than the currently loaded.**
- **A hardware identifications provided in the Security_Configuration files shall be verified against a unique hardware identification, digitally stored in the TOE.**

If the signature, version number, or hardware identification verification fails, the update shall be rejected and an audit event shall be sent.

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

a) At first start-up after delivery, initial values of Security_Configuration and Dynamic_Parameters shall be provided as factory setting. It shall be possible to load initial Security_Configuration files including Administrator's certificate and information required for authentication at the Administration Node. The CA that has signed the initial Security_Configuration files shall be accepted as trusted.

b) The only way to retrieve the Security_Configuration and Dynamic_Parameters factory settings shall be by turning an, optional, physical factory settings switch accessible only after breaking the tamper shield and opening the lid. If the TOE is not equipped with a factory settings switch the factory settings are not retrievable.

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

a) The TSF shall deny injecting Security_Configuration files and modification of Dynamic_Parameters in all states other than Configuration State. The Configuration State shall only be accessible after turning a mandatory physical configuration switch. The Security_Configuration, Dynamic_Parameters, and TOE_TSF shall reside in persistent memory and be writing protected in all states other than Configuration state.

b) The TSF shall deny injection of Security_Configuration files and modification of Dynamic_Parameters from all interfaces other than the administration interface reachable through an Administration node.

Dependencies: FDP_ACC.1– Subset access control
FMT_MSA.3– Static attribute initialisation

Application note: A user data protection method, FDP_ACF, is used here for TSF data.

6.1.4.3 Data authentication with identity of guarantor, FDP_DAU.2

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

FDP_DAU.1.2 The TSF shall provide **the Filter Node** with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

Dependencies: No dependencies.

6.1.4.4 Subset information flow control, FDP_IFC.1a (Audit_Data)

FDP_IFC.1.1a The TSF shall enforce the **Audit_Data information flow control policy** on **the Service Nodes and Filter Node when Audit_Data is generated, transferred within, temporarily stored, and exported from the TOE.**

Dependencies: FDP_IFF.1 Simple security attributes

6.1.4.5 Subset information flow control, FDP_IFC.1b (Traffic_Data)

FDP_IFC.1.1b The TSF shall enforce the **Traffic_Data information flow control policy** on **the Service Nodes and Filter Node when Traffic_Data is transferred through the TOE.**

Dependencies: FDP_IFF.1 Simple security attributes

6.1.4.6 Simple security attributes, FDP_IFF.1a (Audit_Data)

FDP_IFF.1.1a The TSF shall enforce the **Audit_Data information flow control policy** based on the following types of subject and information security attributes:

Subjects and attributes:
Service and Filter Nodes – Their TOE internal addressing identification

Information and attributes:
Audit_Data – Message identification and integrity protection appendix

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

a) Audit_Data generated in Service Nodes shall be transferred to the Filter_Node.

b) Audit_Data shall be integrity protected if temporarily stored within the TOE. Only Audit_Data that has been integrity verified to be correct shall be transferred to the Filter_Node.

c) The availability of Audit_Data shall be guaranteed if temporarily stored within the TOE.

d) Audit_Data shall be exported to an external log system from the Filter_Node.

FDP_IFF.1.3a The TSF shall enforce the **additional information flow control SFP rule**: **None.**

FDP_IFF.1.4a The TSF shall explicitly authorise an information flow based on the following rules: **None**.

FDP_IFF.1.5a The TSF shall explicitly deny an information flow based on the following rules: **None**.

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

Application note: Audit_Data shall be sent using a SysLog, or optional, a SysLog-NG format to an external log server.

6.1.4.7 Simple security attributes, FDP_IFF.1b (Traffic_Data)

FDP_IFF.1.1b The TSF shall enforce the **Traffic_Data information flow control policy** based on the following types of subject and information security attributes:

Subjects and attributes:

Service Nodes – The TOE internal addressing identification,
rules for a well-defined intermediary data format, and
current TOE state

Filter Node – The TOE internal addressing identification,
definition of a well-defined intermediary data format,
definition of data syntax rules,
definition of an object model,
definition of data filter rules,
Authorised_Senders' certificates,
CA certificate, and
current TOE state.

Information and attributes:

Traffic_Data – Meta data for messages and files, and digital signature.

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

a) The intermediary data format used for intra-TOE communication between the Filter and the Service nodes shall be verified to be correctly applied.

b) The Traffic_Data syntax shall be validated to be correct.

c) The Traffic_Data shall comply with the data filter rules, expressed in a Turing complete filter language.

d) If specified by the filter rules, a digital signature supplied to the data by the Authorised_Sender shall be verified against a CA certificate trusted for Traffic_Data approval. The data shall be discarded if the verification fails.

FDP_IFF.1.3b The TSF shall enforce the **additional information flow control SFP rule**: **The TSF shall permit a transfer of Traffic_Data through the TOE network interfaces in Operational State only.**

FDP_IFF.1.4b The TSF shall explicitly authorise an information flow based on the following rules: **None**.

FDP_IFF.1.5b The TSF shall explicitly deny an information flow based on the following rules: **All data transfer through the network interfaces shall be physically prevented in all states other than Operational State**.

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

6.1.4.8 No illicit information flows, FDP_IFF.5

FDP_IFF.5.1 The TSF shall ensure that no illicit information flows exist to circumvent **the Traffic_Data information flow control policy**.

Dependencies: FDP_IFC.1 Subset information flow control

6.1.4.9 Import of user data with security attributes, FDP_ITC.2

FDP_ITC.2.1 The TSF shall enforce the **Configuration access control policy** when importing user data, controlled under the SFP, from outside of the TOE.

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

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

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

FDP_ITC.2.5 The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: [assignment: additional importation control rules].

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FTP_ITC.1 Inter-TSF trusted channel, or
FTP_TRP.1 Trusted path]
FPT_TDC.1 Inter-TSF basic TSF data consistency

Application Note: The import concerns Security_Configuration files according to the Configuration access control policy. The Security_Configuration files may contain CA, Configurator, Administrator, and Authorised_Sender certificates. The standard X.509 v3 shall be supported for the certificates.

A user data protection method is used here for TSF data.

6.1.5 Identification and authentication, FIA

6.1.5.1 User attribute definition, FIA_ATD.1

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: **User id, certificate id[, assignment: list of security attributes]**.

Dependencies: No dependencies.

Application note: The ST author may add other security attributes.

6.1.5.2 Timing of authentication, FIA_UAU.1

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

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

Dependencies: FIA_UID.1 Timing of identification

6.1.5.3 User identification before any action, FIA_UID.2

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.

Dependencies: No dependencies.

6.1.6 Security management, FMT

6.1.6.1 Management of security functions behaviour, FMT_MOF.1

FMT_MOF.1.1 The TSF shall restrict the ability to **modify the behaviour of** the functions

- **Conversion to the intermediary data format used between the Filter and the Service nodes**
- **data syntax verification,**
- **object model conversion,**
- **data filtering,**
- **audit events and data configuration,**
- **trusted CA for Administrators, and**
- **trusted CA for Authorised_Senders**

through the Security_Configuration files to **the Configurator**.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

6.1.6.2 Secure security attributes, FMT_MSA.2

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for **attributes associated with Dynamic_Parameters and Security_Configuration**.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

Application note: Attributes associated with Dynamic_Parameters and Security_Configuration refer to their initial values (factory setting).

6.1.6.3 Management of TSF data, FMT_MTD.1a (Dynamic_Parameters)

FMT_MTD.1.1a The TSF shall restrict the ability to **modify** the **Dynamic_Parameters** to **the Administrator**.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

6.1.6.4 Management of TSF data, FMT_MTD.1b (TOE_Status)

FMT_MTD.1.1b The TSF shall restrict the ability to **query** the

- **the TOE unit hardware identification,**
- **the version number of TOE software parts,**
- **the version number of Security_Configuration files,**
- **the values of Dynamic_Parameters, and**
- **other TOE status values**

to **the Administrator**.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

Application note: The ST author may complement the list of TOE status entities available for the Administrator.

6.1.6.5 Management of TSF data, FMT_MTD.1c (Security_Configuration)

FMT_MTD.1.1c The TSF shall restrict the ability to **modify** the **Security_Configuration** to **the Configurator**.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

6.1.6.6 Revocation, FMT_REV.1

FMT_REV.1.1 The TSF shall restrict the ability to revoke **certificates** associated with the **CA, Configurator, Administrator and Authorised_Sender** under the control of the TSF to the **Administrator**.

FMT_REV.1.2 The TSF shall enforce the rules:

a) **Revocation shall be based on Revocation_Data published by a trusted CA.**

b) **It shall be possible to import Revocation_Data manually and automatically with configurable intervals into the TOE.**

c) **Imported Revocation_Data shall be authentication protected with guarantee of the originator by a digital signature applied by a trusted CA.**

Dependencies: FMT_SMR.1 Security roles

6.1.6.7 Specification of management functions, FMT_SMF.1

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

The Administrator shall be capable to:

a) **Inject Security_Configuration files.**

b) **Modify Dynamic_Parameters.**

c) **Query TOE status.**

d) **Perform state transitions to Configuration State by a physical switch.**

e) **Perform state transitions from Configuration State to Operational State by a physical switch.**

f) **Perform state transition from Initial State to Operational state by a command from the Administrator.**

g) **Revoke certificates.**

h) **Set the time.**

The Configurator shall be capable to define the content of, and digitally sign, the Security_Configuration files.

Dependencies: No dependencies.

Application note: The ST author may specify other management function capabilities.

6.1.6.8 Security roles, FMT_SMR.1

FMT_SMR.1.1 The TSF shall maintain the roles:

- **Configurator and**
- **Administrator.**

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

Dependencies: FIA_UID.1 Timing of identification

Application note: Users are associated with the roles using certificates.

6.1.7 Protection of the TSF, FPT

6.1.7.1 Failure with preservation of secure state, FPT_FLS.1

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

- **Self test verification failure,**
- **software or hardware errors,**
- **critical high temperatures**
- **[, assignment: list of types of failures in the TSF].**

Dependencies: No dependencies.

Application note: The ST author may specify complementing TSF failure types.

6.1.7.2 Passive detection of physical attack, FPT_PHP.1 (Tamper_seal)

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

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

Dependencies: No dependencies.

Application note: Passive detection of physical attack refers to visual inspection of tamper seal and other signs of manipulation.

6.1.7.3 Notification of physical attack, FPT_PHP.2 (Lid_opening)

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

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

FPT_PHP.2.3 For **the opening of the TOE chassi lid during Operational State**, the TSF shall monitor the devices and elements and notify **the external log system analyser** when physical tampering with the TSF's devices or TSF's elements has occurred.

Dependencies: FMT_MOF.1 Management of security functions behaviour

Application note: Notification of physical attack refers to an active generation of an audit event sent to the external log system for analysis according to A.AUDIT_ANALYSIS

6.1.7.4 Reliable time stamps, FPT_STM.1

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

Dependencies: No dependencies.

Application note: Time is used in the creation of reliable time stamps which are used by the audit function and verification of certificates.

6.1.7.5 Inter-TSF basic TSF data consistency, FPT_TDC.1

FPT_TDC.1.1 The TSF shall provide the capability to consistently interpret **Security_Configuration** when shared between the TSF and another trusted IT product.

FPT_TDC.1.2 The TSF shall use **rules for correct interpretation of:**

- **Security_Configuration file version number.**
- **Hardware identification for the TOE units addressed for the security functionality configuration.**
- **Version numbers for all TOE Software parts.**
- **Definition of the intermediate format used between the Filter and Service Nodes.**
- **Scheme definitions for Traffic_Data syntax validation.**
- **Definition of object model used for validation of the filter rules.**
- **Filter rules for Traffic_Data validation written in a Turing complete language.**
- **Audit events and data configuration.**
- **TOE_TSF software updates.**
- **Other configuration items.**

when interpreting the TSF data from another trusted IT product.

Dependencies: No dependencies.

6.1.7.6 TSF testing, FPT_TST.1

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

FPT_TST.1.2 The TSF shall provide **Administrators** with the capability to verify the integrity of **Security_Configuration and Dynamic_Parameters**.

FPT_TST.1.3 The TSF shall provide **Administrators** with the capability to verify the integrity of **TOE_TSF**.

Dependencies: No dependencies.

Application note: The ST author may specify complementing occasions and capabilities. The integrity verification of TSF data and TSF shall be done during all self tests.

6.2 Security Assurance Requirements

In table below, the SAR components included in package EAL4 augmented by ALC_FLR.1 are listed.

Assurance Class	Assurance Component Name	Component
ADV: Development	Security architecture description	ADV_ARC.1
	Complete semi-functional specification with additional error information	ADV_FSP.4
	Implementation representation of the TSF	ADV_IMP.1
	Semiformal modular design	ADV_TDS.3
AGD: Guidance documents	Operational user guidance	AGD_OPE.1
	Preparative procedures	AGD_PRE.1
ALC: Life-cycle support	Production support, acceptance procedures and automation	ALC_CMC.4
	Development tools CM coverage	ALC_CMS.4
	Delivery procedures	ALC_DEL.1
	Identification of security measures	ALC_DVS.1
	Basic flaw remediation	ALC_FLR.1
	Developer defined life-cycle model	ALC_LCD.1
	Compliance with implementation standards	ALC_TAT.1
ASE: Security Target evaluation	Conformance claims	ASE_CCL.1
	Extended components definition	ASE_ECD.1
	ST introduction	ASE_INT.1
	Security objectives	ASE_OBJ.2
	Derived security requirements	ASE_REQ.2
	Security problem definition	ASE_SPD.1
	TOE summary specification	ASE_TSS.1
ATE: Tests	Analysis of coverage	ATE_COV.2
	Testing: modular design	ATE_DPT.1
	Functional testing	ATE_FUN.1
	Independent testing – sample	ATE_IND.2
AVA: Vulnerability assessment	Methodical vulnerability analysis	AVA_VAN.3

Table 12, Security assurance requirements

6.3 Security Requirements Rationale

6.3.1 Security Functional Requirements Dependencies

The following table provides an analysis of how the direct explicit dependencies are met.

Requirement	Direct explicit dependencies	Dependencies met by	Comment
FAU_ARP.1	FAU_SAA.1	FAU_SAA.1	Resolved
FAU_GEN.1	FPT_STM.1	FPT_STM.1	Resolved
FAU_SAA.1	FAU_GEN.1	FAU_GEN.1	Resolved
FAU_SEL.1	FAU_GEN.1 and FMT_MTD.1	FAU_GEN.1 FMT_MTD.1c	Resolved
FAU_STG.1	FAU_GEN.1	FAU_GEN.1	Resolved
FCO_NRO.1	FIA_UID.1	FIA_UID.2	FIA_UID.2 is hierarchical to FIA_UID.1 and the dependency is therefore resolved.
FCO_NRO.2	FIA_UID.1	FIA_UID.2	FIA_UID.2 is hierarchical to FIA_UID.1 and the dependency is therefore resolved.
FCS_COP.1a	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] and FCS_CKM.4	FDP_ITC.2	Resolved FCS_CKM.4 is not needed since no secret keys are distributed.
FCS_COP.1b	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] and FCS_CKM.4	FDP_ITC.2	Resolved FCS_CKM.4 is not needed since no secret keys are distributed.
FCS_COP.1c	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] and FCS_CKM.4	FDP_ITC.2	Resolved FCS_CKM.4 is not needed since no secret keys are distributed.
FCS_COP.1d	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] and FCS_CKM.4	FDP_ITC.2	Resolved FCS_CKM.4 is not needed since no secret keys are distributed.
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1	Resolved
FDP_ACF.1	FDP_ACC.1 and FMT_MSA.3	FDP_ACC.1	Resolved. FMT_MSA.3 is not needed since the static attributes are the initial values of Dynamic_Parameters and Security_Configuration.
FDP_DAU.2	FIA_UID.1	FIA_UID.2	FIA_UID.2 is hierarchical to FIA_UID.1 and the dependency is therefore

Requirement	Direct explicit dependencies	Dependencies met by	Comment
			resolved.
FDP_IFC.1a	FDP_IFF.1	FDP_IFF.1a	Resolved
FDP_IFC.1b	FDP_IFF.1	FDP_IFF.1b	Resolved
FDP_IFF.1a	FDP_IFC.1 and FMT_MSA.3	FDP_IFC.1a	Resolved FMT_MSA.3 is not needed since the static attributes are the initial values of Dynamic_Parameters and Security_Configuration.
FDP_IFF.1b	FDP_IFC.1 and FMT_MSA.3	FDP_IFC.1b	Resolved. FMT_MSA.3 is not needed since the static attributes are the initial values of Dynamic_Parameters and Security_Configuration.
FDP_IFF.5	FDP_IFC.1	FDP_IFC.1b	Resolved
FDP_ITC.2	[FDP_ACC.1 or FDP_IFC.1] and [FTP_ITC.1 or FTP_TRP.1] and FPT_TDC.1	FDP_ACC.1 FPT_TDC.1	Resolved FTP_ITC.1 or FTP_TRP.1 are not needed since the administration interface is protected by physical means.
FIA_ATD.1	-	-	Resolved
FIA_UAU.1	FIA_UID.1	FIA_UID.2	FIA_UID.2 is hierarchical to FIA_UID.1 and the dependency is therefore resolved
FIA_UID.2	-	-	Resolved
FMT_MOF.1	FMT_SMF.1 and FMT_SMR.1	FMT_SMF.1 FMT_SMR.1	Resolved
FMT_MSA.2	[FDP_ACC.1 or FDP_IFC.1] and FMT_SMR.1 and FMT_MSA.1	FDP_ACC.1 FMT_SMR.1	Resolved. FMT_MSA.1 is not needed since the initial values of the Dynamic_Parameters and Security_Configuration are not to be managed by any role.
FMT_MTD.1a	FMT_SMF.1 and FMT_SMR.1	FMT_SMF.1 FMT_SMR.1	Resolved
FMT_MTD.1b	FMT_SMF.1 and FMT_SMR.1	FMT_SMF.1 FMT_SMR.1	Resolved

Requirement	Direct explicit dependencies	Dependencies met by	Comment
FMT_MTD.1c	FMT_SMF.1 and FMT_SMR.1	FMT_SMF.1 FMT_SMR.1	Resolved
FMT_REV.1	FMT_SMR.1	FMT_SMR.1	Resolved
FMT_SMF.1	-	-	Resolved
FMT_SMR.1	FIA_UID.1	FIA_UID.2	FIA_UID.2 is hierarchical to FIA_UID.1 and the dependency is therefore resolved
FPT_FLS.1	-	-	Resolved
FPT_PHP.1	-	-	Resolved
FPT_PHP.2	FMT_MOF.1	Not resolved.	No role authorised by the TOE manage the security functions behaviour after a notification of a physical attack.
FPT_RPL.1	-	-	Resolved
FPT_STM.1	-	-	Resolved
FPT_TDC.1	-	-	Resolved
FPT_TST.1	-	-	Resolved

Table 13, Security functional requirements dependencies

6.3.2 Security Assurance Requirements Dependencies

The assurance level chosen for the TOE is EAL4 augmented by ALC_FLR.1. Since all dependencies are met internally by the EAL package the assurance components dependencies within this Protection Profile are resolved.

6.3.3 Security Functional Requirements Coverage

The following table provides a mapping between security functional requirements and objectives, illustrating that each security functional requirement covers at least one objective and that each objective is covered by at least one security functional or assurance requirement.

	O.AUDIT	O.AUDIT_TRANSFER	O.DOMAIN_SEPARATION	O.ERROR_DETECTION	O.PKI	O.QUERY_TOE_STATUS	O.ROBUST_TOE_ACCESS	O.SECURE_CONFIGURATION	O.SELF_TEST	O.TAMPER_DETECTION	O.TOE_PROTECTION	O.TRAFFIC
ADV_ARC.1			X									
FAU_ARP.1				X					X			
FAU_GEN.1	X											
FAU_SAA.1				X					X			
FAU_SEL.1	X											
FAU_STG.1		X										
FCO_NRO.1												X
FCO_NRO.2					X							
FCS_COP.1a					X		X	X				
FCS_COP.1b					X		X	X				
FCS_COP.1c					X							X
FCS_COP.1d					X							
FDP_ACC.1					X		X	X			X	
FDP_ACF.1					X		X	X			X	
FDP_DAU.2								X				
FDP_IFC.1a		X										
FDP_IFC.1b							X					X
FDP_IFF.1a		X										
FDP_IFF.1b							X					X
FDP_IFF.5												X
FDP_ITC.2					X			X				
FIA_ATD.1							X					

	O.AUDIT	O.AUDIT_TRANSFER	O.DOMAIN_SEPARATION	O.ERROR_DETECTION	O.PKI	O.QUERY_TOE_STATUS	O.ROBUST_TOE_ACCESS	O.SECURE_CONFIGURATION	O.SELF_TEST	O.TAMPER_DETECTION	O.TOE_PROTECTION	O.TRAFFIC
FIA_UAU.1							X					
FIA_UID.2	X						X					
FMT_MOF.1	X				X			X				
FMT_MSA.2								X				
FMT_MTD.1a								X				
FMT_MTD.1b						X						
FMT_MTD.1c								X				
FMT_REV.1					X							
FMT_SMF.1						X		X				
FMT_SMR.1							X	X				
FPT_FLS.1				X					X			
FPT_PHP.1										X		
FPT_PHP.2										X		
FPT_STM.1	X				X							
FPT_TDC.1								X				
FPT_TST.1									X			

Table 14, Security functional requirements coverage

6.3.4 Security Functional Requirements Sufficiency

The suitability and sufficiency of the chosen SFRs are demonstrated below.

O.AUDIT

The objective is covered by:

FAU_GEN.1 ensures the generation of audit events.

FAU_SEL.1 ensures that auditable events are configurable.

FIA_UID.2 ensures that the subject ID is bound to the audit record, when applicable, e.g. for events concerning configuration changes.

FMT_MOF.1 ensures that audit events and data can be configured by and only by the Configurator through the Security_Configuration files.

FPT_STM.1 provides a reliable time stamp to be associated with each event that is raised.

O.AUDIT_TRANSFER

FDP_IFC.1a and **FDP_IFF.1a** defines the Audit_Data information flow control policy that regulates the flow and protection of Audit_Data within the TOE. **FDP_IFF.1a** also defines that Audit_Data shall be exported to an external log system from the Filter_Node.

FAU_STG.1 ensures that Audit_Data is protected from unauthorised modification and deletion if temporarily stored within the TOE.

O.DOMAIN_SEPARATION

A strict hardware of the Service Nodes and the Filter Node is ensured by the TOE design assured by ADV_ARC.1.

O.ERROR_DETECTION

FAU_SAA.1 ensures that TOE is able to analyse events from the Service and Filter Nodes concerning hard- or software errors or critical high temperatures. **FAU_ARP.1** ensures that an error message is shown on a display when the specified events occur.

FPT_FLS.1 ensures that a secure state is preserved at failure situations.

O.PKI

FDP_ACC.1 and **FDP_ACF.1** ensures that X.509 v3 CA and user certificates can be injected.

FDP_ITC.2 regulates the import of certificates through Security_Configuration files according to the Configuraion access control policy.

FCS_COP.1a, FCS_COP.1b, FCS_COP.1c, and FCS_COP.1d, ensures verification of digital signatures and certificates.

FMT_REV.1 ensures means to revoke certificates listed in CRL or by OCSP, that revocation data may be imported manually and automatically, and that the revocation data is authentication protected with guarantee of the originator by a digital signature applied by a trusted CA.

FCO_NRO.2 ensures proof of origin can be applied on imported CRL and OCSP data.

FMT_MOF.1 ensures that trusted CA for Administrators and Authorised_Senders can be changed by the Configurator.

FPT_STM.1 provides a reliable time stamp for evaluation of the validity of certificates.

O.QUERY_TOE_STATUS

FMT_SMF.1 ensures that the Administrator can query the TOE status and **FMT_MTD.1b** ensures that only the Administrator can query the TOE status.

O.ROBUST_TOE_ACCESS

FDP_ACC.1 and **FDP_ACF.1** state that only authenticated Administrators shall be authorised to inject Security_Configuration files and modify Dynamic_Parameters.

FDP_ACC.1 and **FDP_ACF.1** state that Configurators shall be authenticated by a digital signature applied on Security_Configuration files and verified against CA certificate.

FDP_ACC.1 and **FDP_ACF.1** state that it shall be possible to import user certificates and identification and authentication information through Security_Configuration files.

FDP_IFC.1b and **FDP_IFF.1b** state that Authorised_Senders may be authenticated by a digital signature applied on Traffic_Data and verified against a CA certificate.

FCS_COP.1a and FCS_COP.1b ensures that verification against digital signatures can be done.

FIA_UID.2 ensures that user identification is done before any other TSF mediated actions may take place.

FIA_UAU.1 ensures that a successful user authentication is performed before any other TSF mediated actions may take place.

FIA_ATD.1 specifies a list of security attributes belonging to users.

FMT_SMR.1 defines the user roles used by the TOE.

O.SECURE_CONFIGURATION

FDP_ACC.1 and **FDP_ACF.1** defines the Configuration access control policy specifying the rules to be applied for configuration and system update of the TOE.

FDP_DAU.2 ensures the data authentication for Security_Configuration files with guarantee of the Configurator as originator.

FCS_COP.11a and FCS_COP.1b ensures that verification against digital signatures can be done.

FDP_ITC.2 regulates the injection of Security_Configuration files according to the Configuratin access control policy and **FPT_TDC.1** ensures the capability to consistently intrerpret Security_Configuration when shared between the TSF and another trusted IT product where the Security_Configuration is produced.

FMT_MOF.1 restrict the ability to disable, enable, and modify the behaviour of TOE security functions to the Configurator through Security_Configuration files.

FMT_MSA.2 ensures the that only secure initial values (factory setting) are used for Dynamic_Parameters and Security_Configuration.

FMT_MTD.1a (Dynamic_Parameters) restricts the ability to modify the Dynamic_Parameters to the Administrator.

FMT_MTD.1c (Security_Configurtion) restricts the ability to modify the Security_Configuration to the Configurator.

FMT_SMF.1 ensures that the TOE is capable of performing the Administrator's management functions.

FMT_SMR.1 defines the Administrator and Configurator roles for the TOE.

O.SELF_TEST

FPT_TST.1 ensures that TOE runs through a defined set of tests of its functionality; at least at start-up and periodically during normal operation.

FPT_FLS.1 ensures that a secure state is preserved at failure situations.

FAU_SAA.1 ensures that TOE is able to analyse events from the Service and Filter Nodes concerning self test verification failure. **FAU_ARP.1** ensures that an error message is shown on a display when the specified event occurs.

O.TAMPER_DETECTION

FPT_PHP.1 ensures a mechanism for passive detection of a physical attack by visual inspection of a tamper seal.

FPT_PHP.2 ensures that the analysing entity of the external log system is notified if the TOE chassi lid is opened during Operational State.

O.TOE_PROTECTION

FDP_ACC.1 and **FDP_ACF.1** regulates that the TOE_TSF, Security_Configuration files and Dynamic_Parameters shall reside in persistent memory and be write protected except in Configuration State.

O.TRAFFIC

FDP_IFC.1b and **FDP_IFF.1b** defines the information flow control policy for Traffic_Data specifying the rules to be applied for Traffic_Data to be passed through the TOE.

FDP_IFF.5 ensures that no illicit information flows exists to circumvent the Traffic_Data information flow control policy.

FCO_NRO.1 ensures that proof of origin can be applied on Traffic_Data.

FCS_COP.1c ensures that Traffic_Data can be protected by a digital signature

6.3.5 Justification of the Chosen Evaluation Assurance Level

The assurance level EAL4 augmented with ALC_FLR.1 has been chosen as appropriate for a content filtering information gateway with the ability to separating two networks since it provides a moderate level of assured security, and a thorough investigation of the TOE.

The strength of the network separation relies on the configurable filter rules complying with the network separation security policy and is not a part of the TOE evaluation.

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), resources and motivation are assumed to be high because of possible high-value assets protected by the TOE but the opportunity for an attack is considered to be limited. The overall attack potential is assumed to be enhanced-basic, which means that EAL4 is considered an appropriate assurance level.

The assurance requirements are not reproduced in this protection profile as they are chosen from Common Criteria Part 3, without any alterations done to the SARs.

APPENDIX A – ABBREVIATIONS AND ACRONYMS

Administrator node	A part of TOE, a stand alone computer, used by the TOE administrator/Security officer for administrative tasks. Example of a task is the injection of new filtering rules into the filtering node.
CA	Certificate Authority. Publishing trusted certificates and public/private keys.
Covert Channel	Illicit and not approved flows of data, that is, objects not allowed to communicate still manages to do so, hidden from the control of the implemented access control mechanisms.
CRL	Certification Revocation List. Basically list containing information about revoked certificates.
CPU	Central Processing Unit.
DNS	Domain Name System
EAL	A package consisting of assurance components from Part 3 that represents a point on the CC predefined assurance scale.
Filter	Filtering rules, rules that are used in the TOE as to verify the structure and content of data and messages handled by the TOE.
Filter node	Part of the TOE responsible for configurable examination of data using filtering rules, etc.
Heartbeat	Periodically generated event notifying that the TOE is still alive.
IEEE	Institute of Electrical and Electronics Engineers.
Integrity protection appendix	Tag for integrity protection mechanisms appended to a defined set of data , e.g. an HASH value.
IP	Internet Protocol.
IPv4 / IPv6	Internet Protocol version 4 resp. version 6.
MAC	Media Access Control sublayer. Part of the second layer, Data Link Layer, of the OSI model.
MTU	Maximum Transmission Unit
OCSP	Online Certificate Status Protocol.

OSI	Open Systems Interconnection.
OSP	One or more security rules, procedures, practices, or guidelines imposed by an organisation upon its operations.
PKI	Public Key Infrastructure.
PP	Protection Profile, an implementation-independent set of security requirements for a category of TOEs that meet specific consumer needs.requirements (SFRs), security assurance requirements (SARs) and rationales.
Service node	The entrance point for the networks to the TOE. The service nodes are connected to the filtering node during data flows between the networks.
SFP	The security policy enforced by a security function.
SPD	Security problem definition, a part of the PP that contains information about the threats, assumptions and the OSP that constitutes the operational environment in which TOE is to work in.
ST	The central document that specifies security evaluation criteria to substantiate the vendor's claims for the product's security properties.
TCP/IP	Transmission Control Protocol / Internet Protocol
TOE	An IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.
Topology	Here, the IS/IT-infrastructure on the connected networks.
Turing complete	A system of data-manipulation rules (such as an instruction set, a programming language, or a cellular automaton) is said to be Turing complete if it can be used to simulate, in principle, any computer.
XML	eXtensible Markup Language