Curtiss-Wright Controls Defense Solutions

VPX3-685 Secure Router

Versions: VPX3-685-A13014, VPX3-685-A13020-FC, VPX3-685-C23014-FC, and VPX3-685-C23020-FC

Security Target

Evaluation Assurance Level (EAL): EAL2+
Document Number: 828052
Document Version: 1.15

Prepared for:

Curtiss-Wright Controls Defense Solutions
333 Palladium Dr.
Kanata, Ontario, K2V 1A6
Canada

Phone: +1 (613) 599-9199
http://www.cwcdefense.com

Prepared by:

Corsec Security, Inc.

13135 Lee Jackson Memorial Highway, Suite 220
Fairfax, VA 22033
United States of America

Phone: +1 (703) 267-6050
http://www.corsec.com

Table of Contents

Table of Figures

List of Tables

1 Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the ST organization. The Target of Evaluation (TOE) is the Curtiss-Wright VPX3-685 Secure Router, and will hereafter be referred to as the TOE throughout this document. The TOE is a single blade that provides secure routing, switching, firewall functionality, intrusion detection and prevention, and support for virtual private networks (VPNs) in a 3U VPX form factor. The TOE is manufactured in multiple physical configurations associated with the following model numbers which are included in the Common Criteria (CC) evaluation:

- VPX3-685-A13014-FC with Software Version 605714-200\REL2.0.0
- VPX3-685-A13020-FC with Software Version 606163-200\REL2.0.0
- VPX3-685-C23014-FC with Software Version 605714-200\REL2.0.0
- VPX3-685-C23020-FC with Software Version 606163-200\REL2.0.0

1.1 Purpose

This ST is divided into nine sections, as follows:

- Introduction (Section 1) – Provides a brief summary of the ST contents and describes the organization of other sections within this document. It also provides an overview of the TOE security functions and describes the physical and logical scope for the TOE, as well as the ST and TOE references.
- Conformance Claims (Section 2) – Provides the identification of any CC, Protection Profile, and Evaluation Assurance Level (EAL) package claims. It also identifies whether the ST contains extended security requirements.
- Security Problem (Section 3) – Describes the threats, organizational security policies, and assumptions that pertain to the TOE and its environment.
- Security Objectives (Section 4) – Identifies the security objectives that are satisfied by the TOE and its environment.
- Extended Components (Section 5) – Identifies new components (extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs)) that are not included in CC Part 2 or CC Part 3.
- Security Requirements (Section 6) – Presents the SFRs and SARs met by the TOE.
- TOE Summary Specification (Section 7) – Describes the security functions provided by the TOE that satisfy the security functional requirements and objectives.
- Rationale (Section 8) - Presents the rationale for the security objectives, requirements, and SFR dependencies as to their consistency, completeness, and suitability.
- Acronyms (Section 9) – Defines the acronyms and terminology used within this ST.

1.2 Security Target and TOE References

Table 1 provides an overview of the Security Target, as well as the official TOE reference and its FIPS 140-2 validation status.

Table 1 – ST and TOE References

ST Title	Curtiss-Wright Controls Defense Solutions VPX3-685 Secure Router Security Target
ST Version	Version 1.15
ST Author	Corsec Security, Inc.
ST Publication Date	October 18, 2013

ST Title	Curtiss-Wright Controls Defense Solutions VPX3-685 Secure Router Security Target
TOE Reference	Curtiss-Wright VPX3-685 Secure Routers, Hardware Version 1.0 rev A, Models: VPX3-685-A13014-FC and VPX3-685-C23014-FC with Software Version 605714-200, Models:VPX3-685-A13020-FC and VPX3-685-C23020-FC with Software Version 606163-200
FIPS 140-2 Status	Level 2, Validated Crypto Module, Certificate #[TBD][1] for Curtiss-Wright VPX3-685 Secure Routers, Hardware Version 1.0 rev A, Models: VPX3-685-A13014-FC and VPX3-685-C23014-FC with Software Version 605714-200, Models:VPX3-685-A13020-FC and VPX3-685-C23020-FC with Software Version 606163-200

1.3 Product Overview

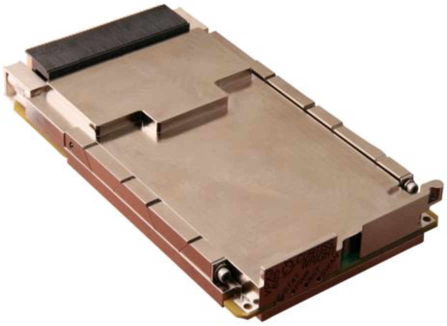
The TOE is a fully-featured Layer 2 and 3 managed Gigabit Ethernet (GbE) router featuring a highly-integrated security sub-system, in a rugged OpenVPX®-compliant 3U VPX[2] card. It speeds and simplifies the integration of secure GbE switching and routing into embedded systems designed for harsh environment military applications. Targeting highly-secure IPv4 and IPv6 Intra-Platform Networks (IPNs), the VPX3-685 is designed to prevent unauthorized access to critical information for applications deployed in air, land, and sea vehicles. It can be used to secure a data storage network or to protect mission critical applications from hostile attacks in the forms of viruses, IP[3] spoofing, denial of service (DoS), and trojan horses. The TOE must be embedded inside a chassis and requires a +5V and/or +3.3Vpower supply.

The VPX3-685 is fixed within two different enclosures: a conduction cooled cover and an air-cooled cover. The internal printed circuit board, the hardware, and software are identical regardless of the enclosure. Curtiss-Wright offers a 14 Ethernet port and a 20 Ethernet port version of each enclosure type. The VPX3-685-A13014-FC and VPX3-685-A13020-FC are the 14 and 20 port air-cooled enclosures, respectively. The VPX3-685-C23014-FC and VPX3-685-C23020-FC are the 14 and 20 port conduction-cooled enclosures, respectively. All of the variants have the identical software load. Separate software part numbers are assigned to facilitate the number of ports exposed to the customer. Figure 1 below shows a pictures of the conduction cooled and the air-cooled VPX3-685 Secure Router.

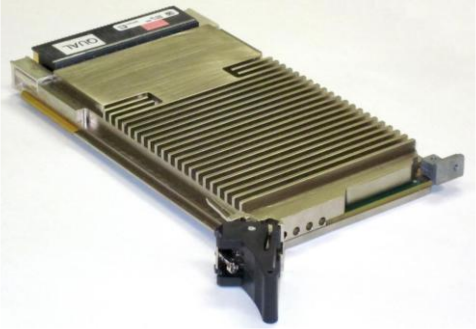
[1] Note to the Lab: The FIPS certificate number will be updated upon completion of the FIPS validation.

[2] VPX: formerly known as "VITA 46", VPX is an ANSI standard (ANSI/VITA 46.0-2007) that provides VMEbus-based systems with support for switched fabrics over a high speed connector.

[3] IP: Internet Protocol



Conduction cooled cover



Air-cooled cover

Figure 1 – VPX3-685 Secure Router

The VPX3-685 can be configured with up to twenty 1GbE[4] interfaces. It also provides up to two 10GbE ports to support switch-to-switch expansion or dual-redundant networks (for fail-over), or for architecting high-performance 10 Gb/s network backbones. Embedded backplane routing is supported with standard Base-T or Base-X GbE and 10GbE XAUI[5] interfaces. To reduce power requirements, any unused GbE ports can be disabled or depopulated. The TOE is available in conduction-cooled, stackable, and air-cooled versions.

The blade incorporates security software and a high-performance hardware-based security engine. Comprising a single-card Unified Threat Management (UTM) system, the blade's advanced security and network features include:

- Support for VPNs[6] (IPSec[7]/PPTP[8]/L2TP[9]) to protect dedicated networks
- A stateful firewall to protect against multiple evasive attacks
- Network Address Translation (NAT) routing for IPv4 masquerading
- Port- and protocol-based Access Control Lists to prevent unauthorized access
- Broadcast Storm Control to protect against network disruption due to packet flooding
- IPv6 with IPSec tunneling for secure communications channels
- Advanced standards-based cryptographic functions (encryption, decryption, and authentication).
- Network traffic analysis and intrusion detection with prevention system (IDS and IPS) functionality

1.4 TOE Overview

The TOE Overview summarizes the usage and major security features of the TOE. The TOE Overview provides a context for the TOE evaluation by identifying the TOE type, describing the TOE, and defining the specific evaluated configuration.

The hardware-software TOE is the VPX3-685 Secure Router. It comprises the VPX3-685 secure router hardware, but does not include the enclosure in which it is installed, nor the power supply to which it is

[4] GbE: Gigabit Ethernet

[5] XAUI: 10 Attachment Unit Interface

[6] VPN: Virtual Private Network

[7] IPSec: Internet Protocol Security

[8] PPTP: Point-to-Point Tunneling Protocol

[9] L2TP: Layer 2 Tunneling Protocol

connected. The TOE also includes the entire VPX3-685 Secure Router software image, and all functions of the software image (excluding the features listed in Section 1.6.3) are included within the TOE boundary[10] and can be configured for use by the TOE administrator. The SFRs in this Security Target make claims about a subset of the VPX3-685 Secure Router's functionality; see Sections 1.6.2 and Section 6 below for details of the specific security function claims.

The VPX3-685 provides several management and configuration interfaces. Remote administrators can connect to a web-based management graphical user interface (GUI) over an HTTPS[11] session or use the CLI remotely over an SSH[12] connection or SNMP[13] v3. The command line interface (CLI) provides a robust CLI environment that uses commands similar to the existing industry standard. The web GUI provides a subset of the commands that are available from the CLI. Through these connections administrators are able to configure and manage switching rules, access control lists (ACLs) and to create or modify the firewall rules to be enforced.

Figure 2 below illustrates the deployment configuration of the TOE.

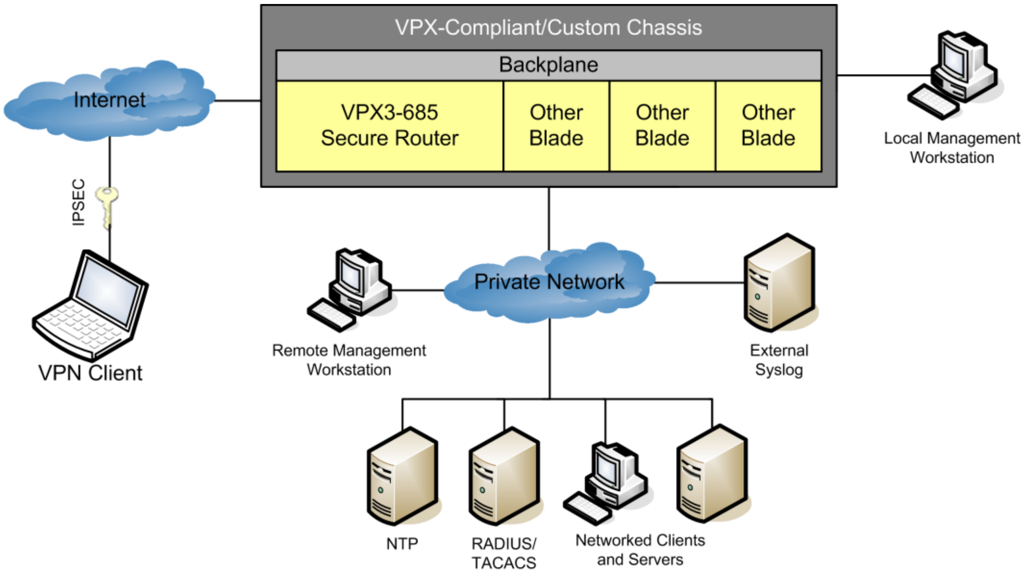


Figure 2 – Deployment Configuration of the TOE

1.5 TOE Environment

The TOE is a single custom 3 or 4U VPX form factor blade running a fully-integrated software solution to provide switch, router, firewall, intrusion detection and prevention system, and VPN functionalities. The TOE must be embedded inside a VPX-compliant/custom chassis and requires a +5V and/or +3.3V power supply.

The TOE stores all audit records in non-volatile memory. Audit records must be transferred to a syslog server in the IT environment for viewing by administrators.

The TOE needs the following environmental components in order to function properly:

- a VPX-compliant/custom chassis

[10] See Section 1.6.3.
[11] HTTPS: HyperText Transfer Protocol Secure
[12] SSH: Secure SHell
[13] SNMP: Simple Network Management Protocol

- a power supply
- an administrator workstation with an SSH/HTTPS client
- an external syslog server

The TOE is intended to be deployed in secure military environments that protect physical access to the TOE. The TOE is intended to be managed by administrators operating under a consistent security policy.

1.6 TOE Description

This section primarily addresses the physical and logical components of the TOE included in the evaluation.

1.6.1 Physical Scope

Figure 3 below illustrates the physical scope and the physical boundary of the overall solution and ties together all of the components of the TOE and the constituents of the TOE Environment.

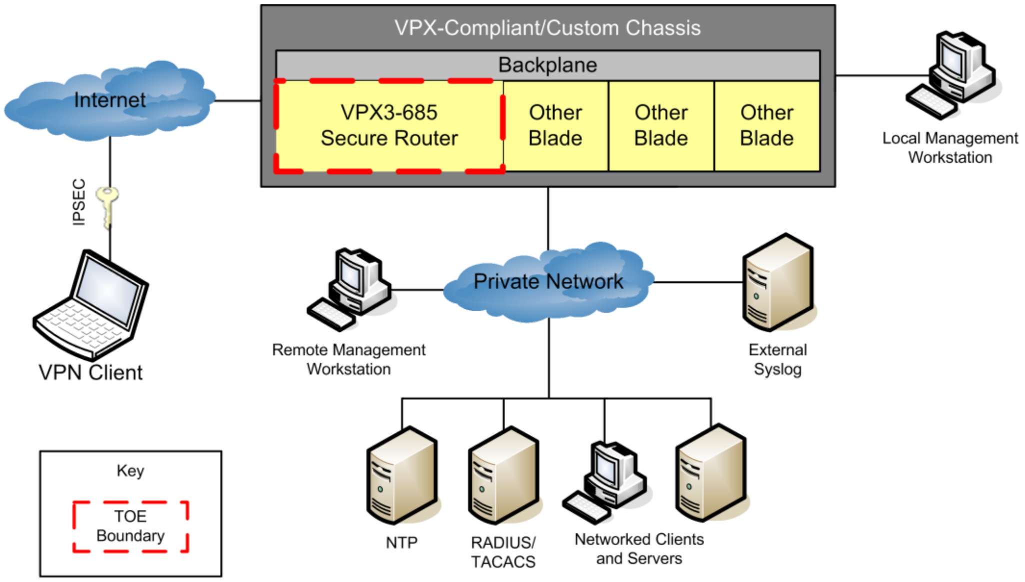


Figure 3 – TOE Physical Boundary

The TOE boundary includes the VPX3-685 hardware, firmware, and software. The TOE boundary does not encompass the VPX-compliant/custom chassis or power supply.

1.6.1.1 TOE Guidance Documentation

The TOE includes the following guidance:

- *VPX3-685 Secure Ethernet Router User's Manual*
- *VPX3-685 WEB Interface Software Reference Manual*
- *VPX3-685 Command Line Interface (CLI) Software Reference Manual*
- *68x Controlled Information User Manual*
- *VPX3-685 Product Release Notes*

1.6.2 Logical Scope

The logical boundary of the TOE will be broken down into the following security classes which are further described in sections 6 and 7 of this ST. The logical scope also provides the description of the security features of the TOE. The security functional requirements implemented by the TOE are usefully grouped under the following TOE Security Functions (TSFs):

- Intrusion Detection
- Protected Communications
- Residual Information Clearing
- Resource Availability
- System Monitoring
- TOE Administration
- Traffic Filter Firewall
- TSF Self Test
- Verifiable Updates

1.6.2.1 Intrusion Detection

The Intrusion Detection function enables the TOE to collect data about network traffic on monitored networks, analyze the collected data for potential statistical and signature-based security violations, automatically react to detected potential security violations, and allow authorized administrators to review the collected data and analyses.

1.6.2.2 Protected Communications

The Protected Communications TSF ensures that the three types of communications in which the TOE participates (TOE to remote administrator; TOE to remote TOE; and TOE to remote IT entity) are secure.

1.6.2.3 Residual Information Clearing

The Residual Information Clearing TSF ensures that data is not accidentally "leaked" into network packets or cryptographic CSPs[14] by ensuring that any data object representing a network packet or CSP is destroyed when that data object is no longer needed.

1.6.2.4 Resource Availability

The Resource Availability TSF ensures that the TOE's resources supporting the administrative interfaces are not exhausted (causing failure of the TOE) by enforcing a quota on the number of simultaneous administrative connections.

1.6.2.5 System Monitoring

The System Monitoring TSF generates audit data, ensuring that sufficient information exists to allow Security Administrators to discover both intentional and unintentional problems with the configuration or operation of the TOE, and that the audit data is protected from compromise.

1.6.2.6 TOE Administration

The TOE Administration TSF provides a trusted means for administrators to interact with the TOE for management purposes via the TOE's web GUI secured via the HTTPS[15] protocol, CLI protected via the SSH protocol, or SNMP v3[16] protocol. These interfaces are protected to mitigate threats of administrator impersonation, account compromise, or accidental access by unwitting users.

1.6.2.7 Traffic Filter Firewall

The Traffic Filter Firewall TSF defines and enforces the UNAUTHENTICATED information flow control Security Functional Policy (SFP). This SFP ensures that the TOE mediates all attempts by unauthenticated external IT entities to send and receive data through the TOE to each other.

[14] CSP: Critical Security Parameter

[15] HTTPS: Secure Hypertext Transport Protocol

[16] SNMP: Simple Network Management Protocol

1.6.2.8 TSF Self Test

The TSF Self Test TSF ensures that the TOE verifies the correct operation of critical TOE functions at power on.

1.6.2.9 Verifiable Updates

The Verifiable Updates TSF enables the administrator to ensure that software or firmware updates are unmodified and are authentic before installation.

1.6.3 Product Features and Functionality Not Included in the TSF

All product functionality and features of the VPX3-685 are included in the TSF.

2 Conformance Claims

This section provides the identification for any CC, Protection Profile (PP), and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for CC and PP conformance claims can be found in Section 8.1.

Table 2 – CC and PP Conformance

Common Criteria (CC) Identification and Conformance	*Common Criteria for Information Technology Security Evaluation*, Version 3.1, Revision 3, July 2009; CC Part 2 extended; CC Part 3 conformant; Parts 2 and 3 Interpretations of the Common Evaluation Methodology (CEM) as of July 22, 2011 were reviewed, and no interpretations apply to the claims made in this ST.
PP Identification	*None*
Evaluation Assurance Level	EAL2+ augmented with Flaw Remediation (ALC_FLR.2)

3 Security Problem

This section describes the security aspects of the environment in which the TOE will be used and the manner in which the TOE is expected to be employed. It provides the statement of the TOE security environment, which identifies and explains all:

- Known and presumed threats countered by either the TOE or by the security environment
- Organizational security policies with which the TOE must comply
- Assumptions about the secure usage of the TOE, including physical, personnel and connectivity aspects

3.1 Threats to Security

This section identifies the threats to the IT[17] assets against which protection is required by the TOE or by the security environment. The threat agents are divided into two categories:

- Attackers who are not TOE users: They have public knowledge of how the TOE operates and are assumed to possess a low skill level, limited resources to alter TOE configuration settings or parameters and no physical access to the TOE.
- TOE users: They have extensive knowledge of how the TOE operates and are assumed to possess a high skill level, moderate resources to alter TOE configuration settings or parameters and physical access to the TOE. (TOE users are, however, assumed not to be willfully hostile to the TOE.)

Both are assumed to have a low level of motivation. The IT assets requiring protection are the TSF[18] and user data saved on or transitioning through the TOE and the hosts on the protected network. Removal, diminution and mitigation of the threats are through the objectives identified in Section 4 Security Objectives. The following threats are applicable:

Table 3 – Threats

Name	Description
T.ADMIN_ERROR	An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.
T.ASPOOF	An unauthorized person may carry out spoofing in which information flows through the TOE into a connected network by using a spoofed source address.
T.FALACT	An attacker might introduce identified or suspected vulnerabilities or perform inappropriate activity to which the TOE fails to react. [19]
T.FALASC	An attacker might introduce vulnerabilities or perform inappropriate activity which the TOE fails to identify based on association of IDS data received from all data sources. [19]
T.FALREC	An attacker might introduce vulnerabilities or perform inappropriate activity which the TOE fails to recognize based on IDS data received from each data source. [19]
T.INADVE	Inadvertent activity and access by attackers who are not TOE users may occur on an IT System the TOE monitors.[19]

[17] IT – Information Technology

[18] TSF – TOE Security Functionality

Name	Description
T.MEDIAT	An unauthorized person may send impermissible information through the TOE which results in the exploitation of resources on the internal network.
T.MISACT	Malicious activity by attackers who are not TOE users, such as introductions of Trojan horses and viruses, may occur on an IT System the TOE monitors.[19]
T.MISUSE	Unauthorized accesses and activity (by attackers who are not TOE users) indicative of misuse may occur on an IT System the TOE monitors.[19]
T.RESOURCE_EXHAUSTION	A process initiated by a TOE user, or a TOE user may deny access to TOE services by exhausting critical resources on the TOE.[19]
T.SCNCFG	Improper security configuration settings created by non-TOE users may exist in the IT System the TOE monitors.[19]
T.SCNMLC	Users could execute malicious code on an IT System that the TOE monitors which causes modification of the IT System protected data or undermines the IT System security functions.
T.SCNVUL	Vulnerabilities (introduced by non-TOE users) may exist in the IT System the TOE monitors.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.

3.2 Organizational Security Policies

An Organizational Security Policy (OSP) is a set of security rules, procedures, or guidelines imposed by an organization on the operational environment of the TOE. The following OSPs are presumed to be imposed upon the TOE or its operational environment by any organization implementing the TOE in the CC evaluated configuration:

Table 4 – Organizational Security Policies

Name	Description

Name	Description
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

3.3 Assumptions

This section describes the security aspects of the intended environment for the evaluated TOE. The operational environment must be managed in accordance with assurance requirement documentation for delivery, operation, and user guidance. The following specific conditions are required to ensure the security of the TOE and are assumed to exist in an environment where this TOE is employed.

Table 5 – Assumptions

Name	Description
A.NO_GENERAL_PURPOSE	It is assumed that there are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
A.TRUSTED NETWOK_BOUNDARY	The TOE router controls the single access point to the trusted network and that there are no hostile entities on the trusted network side.

4 Security Objectives

Security objectives are concise, abstract statements of the intended solution to the problem defined by the security problem definition (see Section 3). The set of security objectives for a TOE form a high-level solution to the security problem. This high-level solution is divided into two part-wise solutions: the security objectives for the TOE, and the security objectives for the TOE's operational environment. This section identifies the security objectives for the TOE and its supporting environment.

4.1 Security Objectives for the TOE

The specific security objectives for the TOE are as follows:

Table 6 – Security Objectives for the TOE

Name	Description
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.IDANLZ	The TOE must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).
O.IDSCAN	The TOE must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.
O.IDSENS	The TOE must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.
O.MEDIAT	The TOE must mediate the flow of all information from users on a connected network to users on another connected network.
O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.RESOURCE_AVAILABILITY	The TOE shall provide mechanisms that mitigate user attempts to exhaust TOE resources (e.g., persistent storage).
O.RESPON	The TOE must respond appropriately to analytical conclusions.
O.SECSTA	Upon initial start-up of the TOE or recovery from an interruption in TOE service, the TOE must not compromise its resources or those of any connected network.
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.

O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.

4.2 Security Objectives for the Operational Environment

4.2.1 IT Security Objectives

The following IT security objectives are to be satisfied by the environment:

Table 7 – IT Security Objectives

Name	Description
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
OE.TRUSTED NETWOK_BOUNDARY	The TOE router controls the single access point to the trusted network and that there are no hostile entities on the trusted network side.

4.2.2 Non-IT Security Objectives

There are no non-IT environment security objectives.

5 Extended Components

This section defines the extended SFRs and extended SARs met by the TOE. These requirements are presented following the conventions identified in Section 6.1.

5.1 Extended TOE Security Functional Components

This section specifies the extended SFRs for the TOE. The extended SFRs are organized by class. Table 8 identifies all extended SFRs implemented by the TOE

Table 8 – Extended TOE Security Functional Requirements

Name	Description
FAU_STG_EXT.1	External Audit Trail Storage
FCS_CKM_EXT.4	Cryptographic key destruction
FCS_COMM_PRO_EXT.1	Communication Protection
FCS_HTTPS_EXT.1	HTTPS
FCS_IPSEC_EXT.1	IPsec
FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
FCS_SSH_EXT.1	SSH
FCS_TLS_EXT.1	TLS[20]
FIA_PMG_EXT.1	Password Management
FIA_UAU_EXT.5	Password-based Authentication Mechanism
FIA_UIA_EXT.1	User Identification and Authentication
FPT_PTD_EXT.1	Management of TSF Data
FPT_TST_EXT.1	TSF self test
FPT_TUD_EXT.1	Trusted Update
FTA_SSL_EXT.1	TSF-initiated session locking
IDS_ANL_EXT.1	Analysis
IDS_RCT_EXT.1	Analyzer react
IDS_RDR_EXT.1	Restricted data review
IDS_SDC_EXT.1	System data collection

[20] TLS: Transport Layer Security

5.1.1 Class FAU: Security Audit

Security auditing involves recognizing, recording, storing, and analyzing information related to security relevant activities (i.e. activities controlled by the TSF). The extended family “FAU_STG_EXT: External audit trail” was modeled after the CC family and related components for “FAU_STG: Security audit event storage”.

5.1.1.1 Family FAU_STG_EXT: External audit trail

Family Behaviour

This family defines the requirements for external storage of audit records enforced by the TSF.

Component Leveling

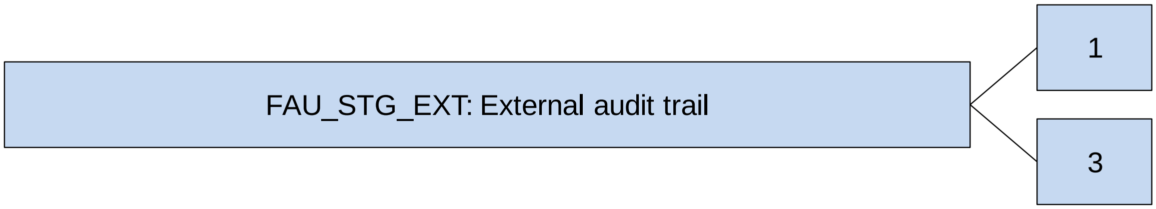


Figure 4 – External audit trail family decomposition

“FAU_STG_EXT.1 External audit trail storage” requires that the TOE at least store its audit trail on an external server, and also support receipt of audit data, over a trusted channel.

Management: FAU_STG_EXT.1

a) No management activities are foreseen.

Audit: FAU_STG_EXT.1

a) No auditable events are foreseen.

FAU_STG_EXT.1 External audit trail storage

Hierarchical to: No other components.

FAU_STG_EXT.1.1

The TSF shall be able to [selection: transmit the generated audit data to an external IT entity over a trusted channel defined in FTP_ITC.1, receive and store audit data from an external IT entity over a trusted channel defined in FTP_ITC.1].

Dependencies: FAU_GEN.1 Audit data generation
FTP_ITC.1(1) Inter-TSF trusted channel (prevention of disclosure)
FTP_ITC.1(2) Inter-TSF trusted channel (prevention of modification)

5.1.2 Class FCS: Cryptographic Support

The TSF may employ cryptographic functionality to help satisfy several high-level security objectives. These include (but are not limited to): identification and authentication, non-repudiation, trusted path, trusted channel and data separation. This class is used when the TOE implements cryptographic functions, the implementation of which could be in hardware, firmware and/or software. The extended family "FCS_CKM_EXT: Cryptographic key management" was modeled after the CC family and related components for "FCS_CKM: Cryptographic key management". The extended families "FCS_COMM_PRO_EXT: Communication protection", "FCS_HTTPS_EXT: HTTPS", "FCS_IPSEC_EXT: IPsec", "FCS_RBG_EXT: Random bit generation", "FCS_SSH_EXT: SSH", and "FCS_TLS_EXT: TLS" were modeled after various families and related components in the CC Class FCS: Cryptographic Support.

5.1.2.1 Family FCS_CKM_EXT: Cryptographic key management

Family Behaviour

Cryptographic keys must be managed throughout their life cycle. This family is intended to support that lifecycle and consequently defines requirements for the following activities: cryptographic key generation, cryptographic key distribution, cryptographic key access and cryptographic key destruction. This family should be included whenever there are functional requirements for the management of cryptographic keys. Components in this family address the requirements for managing cryptographic keys as defined in CC Part 2. This section defines the extended components for the FCS_CKM family.

Component Leveling

FCS_CKM_EXT: Cryptographic key management
4

Figure 5 – Cryptographic key management family decomposition

FCS_CKM_EXT.4 Cryptographic key zeroization, requires cryptographic keys and cryptographic critical security parameters to be zeroized. It was modeled after FCS_CKM.4

Management: FCS_CKM_EXT.4

a) There are no management activities foreseen.

Audit: FCS_CKM_EXT.4

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

a) Failure on invoking the cryptographic key zeroization functionality.

FCS_CKM_EXT.4 Cryptographic Key Zeroization

Hierarchical to: No other components

FCS_CKM_EXT.4.1

The TSF shall zeroize all plaintext secret and private cryptographic keys and CSP[21]s when no longer required.

**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]**

[21] Critical Security Parameters

5.1.2.2 Family FCS_COMM_PROT_EXT: Communications protection

Family Behaviour

Components in this family address the requirements for protecting network communications. This is a new family defined for the FCS Class.

Component Leveling

Figure 6 – Communications protection family decomposition

FCS_COMM_PROT_EXT.1 Communications Protection, requires that network communications be protected. The communications must be protected by either IPSec, SSH, or both. Additionally, TLS/HTTPS may be selected.

Management: FCS_COMM_PROT_EXT.1

a) There are no management activities foreseen.

Audit: FCS_COMM_PROT_EXT.1

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

a) There are no auditable events foreseen.

FCS_COMM_PROT_EXT.1 Communications Protection

Hierarchical to: No other components

FCS_COMM_PROT_EXT.1.1

The TSF shall protect communications using [selection: IPsec, SSH] and [selection: TLS/HTTPS, no other protocol].

Dependencies: [FCS_IPSEC_EXT.1 IPSEC or
FCS_SSH_EXT.1 SSH]
FCS_HTTPS_EXT.1 HTTPS, if selected
FCS_TLS_EXT.1 TLS.

5.1.2.3 Family FCS_HTTPS_EXT: HTTPS

Family Behaviour

Components in this family address the requirements for protecting communications using HTTPS. This is a new family defined for the FCS Class.

Figure 7 – HTTPS family decomposition

FCS_HTTPS_EXT.1 HTTPS, requires that HTTPS be implemented.

Management: FCS_HTTPS_EXT.1

a) There are no management activities foreseen.

Audit: FCS_HTTPS_EXT.1

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

a) Failure to establish an HTTPS session.
a) Establishment/termination of an HTTPS session

FCS_HTTPS_EXT.1 HTTPS

Hierarchical to: No other components

FCS_HTTPS_EXT.1.1

The TSF shall implement the HTTPS protocol that complies with RFC[22] 2818.

FCS_HTTPS_EXT.1.2

The TSF shall implement the HTTPS protocol using TLS as specified in FCS_TLS_EXT.1.

Dependencies: FCS_TLS_EXT.1 TLS

5.1.2.4 Family FCS_IPSEC_EXT: IPsec

Family Behaviour

Components in this family address the requirements for protecting communications using IPSEC. This is a new family defined for the FCS Class.

Component Leveling

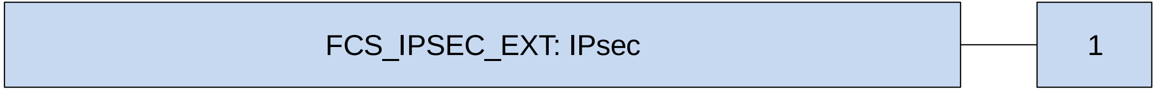


Figure 8 – IPsec family decomposition

FCS_IPSEC_EXT.1 IPsec, requires that IPsec be implemented as specified.

Management: FCS_IPSEC_EXT.1

a) There are no management activities foreseen.

Audit: FCS_IPSEC_EXT.1

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

a) Failure to establish an IPsec SA.
a) Establishment/termination of an IPsec SA.

FCS_IPSEC_EXT.1 IPsec

Hierarchical to: No other components

FCS_IPSEC_EXT.1.1

The TSF shall implement IPsec using the ESP[23] protocol as defined by RFC 4303 using the cryptographic algorithms AES-CBC-128, AES-CBC-256 (both specified by RFC 3602), [selection: no other algorithms] and using IKEv1 as defined in RFCs 2407, 2408, 2409, and RFC 4109; [selection: no other method] to establish the security association.

FCS_IPSEC_EXT.1.2

The TSF shall ensure that IKEv1 Phase 1 exchanges use only main mode.

FCS_IPSEC_EXT.1.3

The TSF shall ensure that IKEv1 SA[24] lifetimes are able to be limited to 24 hours for Phase 1 SAs and 8 hours for Phase 2 SAs.

FCS_IPSEC_EXT.1.4

The TSF shall ensure that IKEv1 SA lifetimes are able to be limited to [assignment: *number between 100 - 200*] MB[25] of traffic for Phase 2 SAs.

FCS_IPSEC_EXT.1.5

The TSF shall ensure that all IKE[26] protocols implement DH[27] Groups 14 (2048-bit MODP), and [selection: 24 (2048-bit MODP with 256-bit POS), 19 (256-bit Random ECP), 20 (384-bit 47 Random ECP), [assignment: *other DH groups that are implemented by the TOE*], no other DH groups].

[22] RFC: Request for Comment
[23] ESP: Encapsulating Security Payload
[24] SA: Security Association
[25] MB: Megabyte
[26] IKE: Internet Key Exchange
[27] DH: Diffie Hellman

FCS_IPSEC_EXT.1.6

The TSF shall ensure that all IKE protocols implement Peer Authentication using the [selection: DSA, rDSA, ECDSA[28]] algorithm.

FCS_IPSEC_EXT.1.7

The TSF shall support the use of pre-shared keys (as referenced in the RFCs) for use in authenticating its IPsec connections.

FCS_IPSEC_EXT.1.8

The TSF shall support the following:

1. Pre-shared keys shall be able to be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")");

2. Pre-shared keys of 22 characters and [selection: [assignment: *other supported lengths*], no other lengths].

Dependencies: FCS_COP.1 Cryptographic operation

5.1.2.5 Family FCS_RBG_EXT: Random bit generation

Family Behaviour

Components in this family address the requirements for random number / bit generation. This is a new family defined for the FCS Class.

Component Leveling

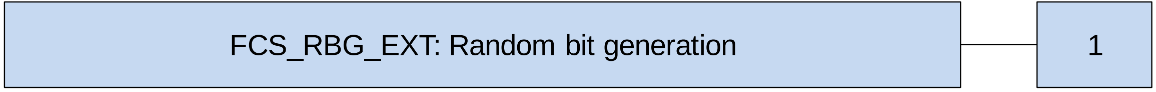


Figure 9 – Random bit generation family decomposition

FCS_RBG_EXT.1 Random Bit Generation, requires random bit generation to be performed in accordance with selected standards and seeded by an entropy source. It was modeled after FCS_COP.1.

Management: FCS_RBG_EXT.1

a) There are no management activities foreseen.

Audit: FCS_RBG_EXT.1

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

a) Failure of the randomization process.

FCS_RBG_EXT.1 Random bit generation

Hierarchical to: No other components

FCS_RBG_EXT.1.1

The TSF shall perform all random bit generation (RBG) services in accordance with [selection, choose one of: NIST[29] Special Publication 800-90 using [selection: Hash_DRBG (any), HMAC_DRBG (any), CTR_DRBG (AES), Dual_EC_DRBG (any)]; FIPS[30] PUB[31] 140-2 Annex C: X9.31 Appendix 2.4 using AES] seeded by an entropy source that accumulated entropy from at least one independent TSF-hardware-based noise source.

[29] NIST: National Institute of Standards and Technology

[29] NIST: National Institute of Standards and Technology

[30] FIPS: Federal Information Processing Standard

[31] PUB: Publication

FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded with a minimum of [selection, choose one of: 128 bits, 256 bits] of entropy at least equal to the greatest bit length of the keys and authorization factors that it will generate.

Dependencies: None.

5.1.2.6 Family FCS_SSH_EXT: SSH

Family Behaviour

Components in this family address the requirements for protecting communications using SSH. This is a new family defined for the FCS Class.

Component Leveling

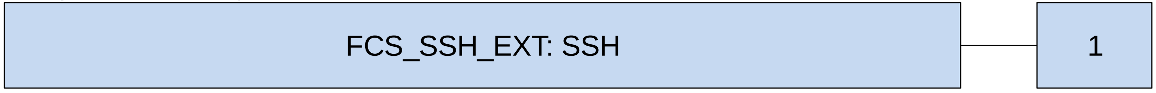


Figure 10 – SSH family decomposition

FCS_SSH_EXT.1 SSH requires that SSH be implemented.

Management: FCS_SSH_EXT.1

There are no management activities foreseen.

Audit: FCS_SSH_EXT.1

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

a) Failure to establish an SSH session.
b) Establishment/termination of an SSH session..

FCS_SSH_EXT.1 SSH

Hierarchical to: No other components

FCS_SSH_EXT.1.1

The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, and 4254.

FCS_SSH_EXT.1.2

The TSF shall ensure that the SSH connection be rekeyed after no more than 2^{28} packets have been transmitted using that key.

FCS_SSH_EXT.1.3

The TSF shall ensure that the SSH protocol implements a timeout period for authentication as defined in RFC 4252 of [assignment: *timeout period*], and provide a limit to the number of failed authentication attempts a client may perform in a single session to [assignment: *maximum number of attempts*] attempts.

FCS_SSH_EXT.1.4

The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, password-based.

FCS_SSH_EXT.1.5

The TSF shall ensure that, as described in RFC 4253, packets greater than [assignment: *number of bytes*] bytes in an SSH transport connection are dropped.

FCS_SSH_EXT.1.6

The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms: AES-CBC-128, AES-CBC-256, [selection: AEAD_AES_128_GCM, AEAD_AES_256_GCM, no other algorithms]**.**

FCS_SSH_EXT.1.7

The TSF shall ensure that the SSH transport implementation uses SSH_RSA and [selection: PGP-SIGN-RSA, PGP-SIGN-DSS, no other public key algorithms] as its public key algorithm(s).

FCS_SSH_EXT.1.8

The TSF shall ensure that data integrity algorithms used in SSH transport connection is [selection: hmac-sha1, hmac-sha1-96, hmac-md5, hmac-md5-96].

FCS_SSH_EXT.1.9

The TSF shall ensure that diffie-hellman-group14-sha1 is the only allowed key exchange method used for the SSH protocol.

Dependencies: FCS_COP.1 Cryptographic operation

5.1.2.7 Family FCS_TLS_EXT: TLS

Family Behaviour

Components in this family address the requirements for protecting communications using TLS. This is a new family defined for the FCS Class.

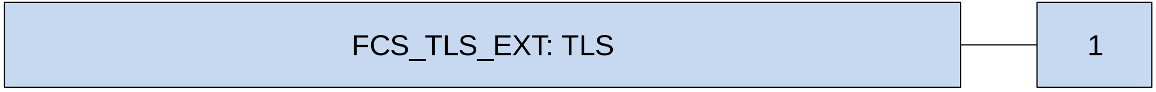


Figure 11 – TLS family decomposition

FCS_TLS_EXT.1 TLS requires that TLS be implemented.

Management: FCS_TLS_EXT.1

There are no management activities foreseen.

Audit: FCS_TLS_EXT.1

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

a) Failure to establish a TLS session.
b) Establishment/termination of a TLS session.

FCS_TLS_EXT.1 TLS

Hierarchical to: No other components

FCS_TLS_EXT.1.1

The TSF shall implement one or more of the following protocols [selection: TLS 1.0 (RFC 2246), TLS 1.1 (RFC 4346), TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA

[selection:
None
TLS_RSA_WITH_AES_128_CBC_SHA256
TLS_RSA_WITH_AES_256_CBC_ SHA256
TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256
TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
].

Dependencies: FCS_COP.1 Cryptographic operation

5.1.3 Class FIA: Identification and Authentication

Families in this class address the requirements for functions to establish and verify a claimed user identity. The extended family “FIA_PMG_EXT: Password management” was modeled after various families and related components in Class FIA. The extended component “FIA_UAU_EXT.5: Password-based authentication mechanism” was modeled after FIA_UAU.5: “Multiple authentication mechanisms”. The extended family “FIA_UIA_EXT.1: User identification and authentication” was modeled after both “FIA_UAU.1: Timing of authentication” and “FIA_UID.1: Timing of identification.”

5.1.3.1 Family FIA_PMG_EXT: Password management

Family Behaviour

This family defines the password complexity requirements that the TSF must enforce and must allow the administrator to configure.

Component Leveling

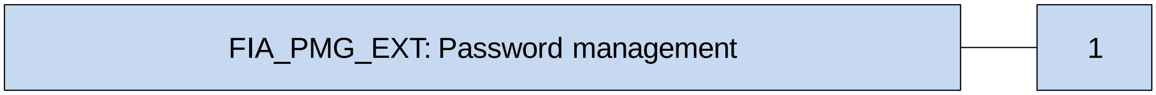


Figure 12 – Password management family decomposition

“FIA_PMG_EXT.1 Password management” defines password complexity requirements and configuration options.

Management: FIA_PMG_EXT.1

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

a) Configure password complexity options

Audit: FIA_PMG_EXT.1

The following actions should be auditable if FAU_GEN.1 Security audit data generation is included in the ST:

a) Configuration of password complexity options

FIA_PMG_EXT.1 Password management

Hierarchical to: No other components

FIA_PMG_EXT.1.1

The TSF shall provide the following password management capabilities for administrative passwords:

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”);
2. Minimum password length shall settable by the Security Administrator, and support passwords of 8 characters or greater;
3. Passwords composition rules specifying the types and number of required characters that comprise the password shall be settable by the Security Administrator.
4. Passwords shall have a maximum lifetime, configurable by the Security Administrator.
5. New passwords must contain a minimum of 4 character changes from the previous password.

Dependencies: No dependencies.

5.1.3.2 Family FIA_UAU_EXT: User authentication

Family Behaviour

This family defines the types of user authentication mechanisms supported by the TSF. This section defines the extended components for the FIA_UAU family.

Component Leveling

Figure 13 – User authentication family decomposition

FIA_UAU_EXT.5 Password-based Authentication Mechanism, requires a local password-based authentication mechanism and the capability for passwords to expire. In addition, other authentication mechanisms can be specified.

Management: FIA_UAU_EXT.5
The following actions could be considered for the management functions in FMT:

a) reset a user password by an administrator.

Audit: FIA_UAU_EXT.5
The following actions should be auditable if FAU_GEN.1 Security audit data generation is included in the ST:

a) All use of the authentication mechanisms.

FIA_UAU_EXT.5 Password-based Authentication Mechanism

Hierarchical to: No other components

FIA_UAU_EXT.5.1

The TSF shall provide a local password-based authentication mechanism, [selection: *[assignment: other authentication mechanism(s)*], none] to perform user authentication.

FIA_UAU_EXT.5.2

The TSF shall ensure that users with expired passwords are [selection: required to create a new password after correctly entering the expired password, locked out until their password is reset by an administrator].

Dependencies: No dependencies.

5.1.3.3 Family FIA_UIA_EXT: User identification and authentication

Family Behaviour
This family defines the services provided by the TSF to unidentified and unauthenticated users.

Component Leveling

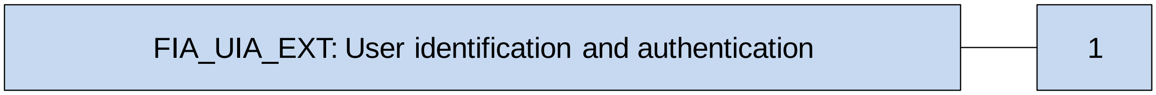


Figure 14 – User identification and authentication family decomposition

FIA_UIA_EXT.1 "User identification and authentication" requires users to be successfully identified and authenticated before providing any services other than those listed to the user.

Management: FIA_UIA_EXT.1

a) There are no management activities foreseen.

Audit: FIA_UIA_EXT.1

a) All use of the identification and authentication mechanism.

FIA_UIA_EXT.1 User Identification and Authentication

Hierarchical to: FIA_UAU.1, FIA_UID.1

FIA_UIA_EXT.1.1

The TSF shall allow [selection:*[assignment: list of TOE-provided services]*, no services] on behalf of the user to be performed before the user is identified and authenticated.

FIA_UIA_EXT.1.2

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

Dependencies: No dependencies.

5.1.4 Class FPT: Protection of the TSF

This class contains families of functional requirements that relate to the integrity and management of the mechanisms that constitute the TSF and to the integrity of TSF data. The extended families "FPT_PTD_EXT: Management of TSF data" and "FPT_TUD_EXT.1: Trusted update" were modeled after various families and related components in Class FPT. The extended component "FPT_TXT_EXT.1: TSF self test" was modeled after "FPT_TST.1: TSF testing".

5.1.4.1 Family FPT_PTD_EXT: Management of TSF data

Family Behaviour

Components in this family address the requirements for managing and protecting TSF data, such as passwords and keys. This is a new family defined for the FPT Class.

Component Leveling

Figure 15 – Management of TSF data family decomposition

FPT_PTD_EXT.1 Management of TSF Data, requires preventing selected TSF data from being read by any user or subject.

Management: FPT_PTD_EXT.1

a) There are no management activities foreseen.

Audit: FPT_PTD_EXT.1

a) There are no auditable activities foreseen.

FPT_PTD_EXT.1(1) Management of TSF data (for reading of authentication data)
Hierarchical to: No other components
FPT_PTD_EXT.1.1

The TSF shall **prevent** reading of *the plaintext passwords.*

Dependencies: No dependencies.

FPT_PTD_EXT.1(2) Management of TSF data (for reading of all symmetric keys)
Hierarchical to: No other components
FPT_PTD_EXT.1.1

The TSF shall **prevent** reading of all *pre-shared keys, symmetric key, and private keys.*

Dependencies: No dependencies.

5.1.4.2 Family FPT_TST_EXT: TSF testing

Family Behaviour
Components in this family address the requirements for self-testing the TSF for selected correct operation.

Component Leveling

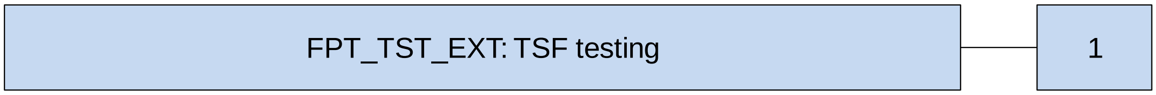


Figure 16 – TSF testing family decomposition

FPT_TST _EXT.1 TSF testing, requires a suite of self tests to be run during initial start-up in order to demonstrate correct operation of the TSF.

Management: FPT_TST_EXT.1

a) There are no management activities foreseen.

Audit: FPT_TST_EXT.1

a) Indication that TSF self-test was completed.

FPT_TST_EXT.1 TSF testing
Hierarchical to: No other components
FPT_TST_EXT.1.1
The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.
Dependencies: No dependencies.

5.1.4.3 Family FPT_TUD_EXT: Trusted update

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

Component Leveling

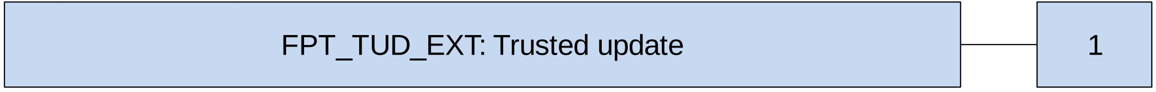


Figure 17 – Trusted update family decomposition

FPT_TUD_EXT.1 Trusted update, requires management tools be provided to update the TOE firmware and software, including the ability to verify the updates prior to installation.

Management: FPT_TUD_EXT.1

a) There are no management activities foreseen.

Audit: FPT_TUD_EXT.1

a) Initiation of update.

FPT_ TUD_EXT.1 Trusted update
Hierarchical to: No other components
FPT_TUD_EXT.1.1
The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.
FPT_TUD_EXT.1.2

The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.3

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

Dependencies: FCS_COP.1 Cryptographic operation.

5.1.5 Class FTA: TOE Access

This family specifies functional requirements for controlling the establishment of a user's session. The extended component “FTA_SSL_EXT.1: TSF-initiated session locking” was modeled after “FTA_SSL.1: TSF_initiated session locking”.

5.1.5.1 Family FTA_SSL_EXT: TSF-initiated Session Locking

Family Behaviour

Components in this family address the requirements for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.

Component Leveling

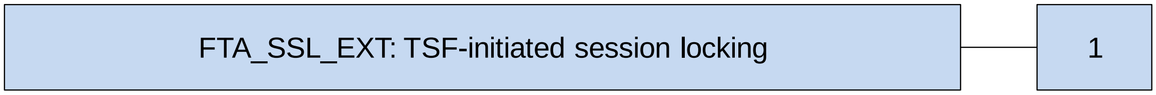


Figure 18 – TSF-initiated Session Locking family decomposition

FTA_SSL_EXT.1 TSF-initiated Session Locking requires system initiated locking of an interactive session after a specified period of inactivity.

Management: FTA_SSL_EXT.1

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

a) Specification of the time of user inactivity after which lock-out occurs for an individual user.

Audit: FTA_SSL_EXT.1

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

a) Any attempts at unlocking an interactive session.

FTA_SSL_EXT.1 TSF-initiated session locking

Hierarchical to: No other components

FTA_SSL_EXT.1.1

The TSF shall, for local interactive sessions, [selection:

- lock the session – disable any activity of the user’s data access display devices other than unlocking the session, and requiring that the administrator re-authenticate to the TSF prior to unlocking the session;
- terminate the session]

after a Security Administrator-specified time period of inactivity.

Dependencies: FIA_UIA_EXT.1 User identification and authentication.

5.1.6 Class IDS: Intrusion Detection Function

Intrusion Detection functions involve collecting information from designated systems and analyzing the information for vulnerabilities and compliance. The extended class "IDS: Intrusion detection function" class was modeled after the CC "FAU: Security audit class". The extended family and related components for "IDS_SDC_EXT: System data collection" were modeled after the CC family and related components for "FAU_GEN: Security audit generation". The extended family "IDS_ANL_EXT: Analyzer analysis" were modeled after the CC family and related components for "FAU_SAA: Security audit analysis". The extended family "IDS_RCT_EXT: Analyzer react" were modeled after the CC family and related components for "FAU_ARP: Security alarms". The extended family and related components for "IDS_RDR_EXT: Restricted data review" were modeled after the CC family and related components for "FAU_STG: Security audit event storage".

5.1.6.1 Family IDS_ANL_EXT: Analyzer analysis

Family Behaviour

This family defines the analysis the TOE performs on the collected system data. This family enumerates the types of program code that shall be collected by the TSF, and identifies what type of control will be enforced on the executable code. This family also determines which changes are to be prevented, and which are to be monitored and reported.

Component Leveling

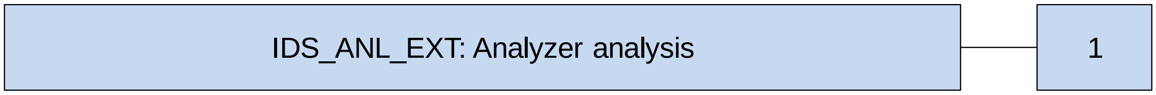


Figure 19 – Analyzer analysis family decomposition

"IDS_ANL_EXT.1 Analyzer analysis" specifies the list of analyses the TOE will perform on the collected system data.

Management: IDS_ANL_EXT.1
The following actions could be considered for the management functions in FMT:

b) Configuration of the analysis to be performed.

Audit: IDS_ANL_EXT.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the ST:

a) Minimal: Enabling and disabling of any of the analysis mechanisms.

IDS_ANL_EXT.1 Analyzer analysis
Hierarchical to: No other components

IDS_ANL_EXT.1.1

The TSF shall perform the following analysis function(s) on all system data collected:

a) [selection: statistical, signature, integrity]; and

b) [assignment: *other analytical functions*].

IDS_ANL_EXT.1.2

The System shall record within each analytical result at least the following information:

a) Date and time of the result, type of result, identification of data source; and

b) [assignment: *other security relevant information about the result*].

Dependencies: IDS_SDC_EXT.1 System Data Collection

FPT_STM.1 Reliable Timestamps

5.1.6.2 Family IDS_RCT_EXT: Analyzer react

Family Behaviour

This family defines the response to be taken in case of detected events indicative of a potential security violation.

Component Leveling

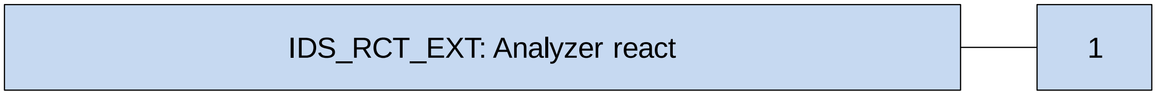


Figure 20 – Analyzer react family decomposition

"IDS_RCT_EXT.1 Analyzer react" specifies actions the TSF shall take in case a potential security violation is detected.

Management: IDS_RCT_EXT.1
The following actions could be considered for the management functions in FMT:
a) the management (addition, removal, or modification) of actions.

Audit: EXT_IDS_RCT.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the ST:
a) Minimal: Actions taken due to a potential security violations.

IDS_RCT_EXT.1 Analyzer react
Hierarchical to: No other components
IDS_RCT_EXT.1.1
The TSF shall send an alarm to [assignment: *alarm destination*] and take [assignment: *appropriate actions*] upon detection of a potential security violation.
Dependencies: IDS_SDC_EXT.1 System data collection

5.1.6.3 Family IDS_RDR_EXT: Restricted data review

Family Behaviour

This family defines the requirements for external storage of audit records enforced by the TSF indicative of a potential security violation.

Component Leveling

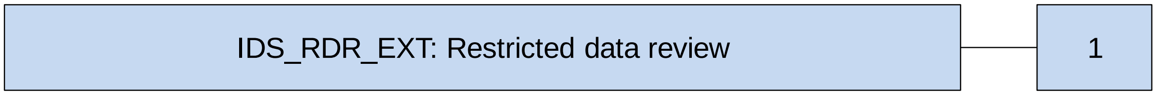


Figure 21 – Restricted data review family decomposition

"IDS_RDR_EXT.1 Restricted data review" requires that the TOE at least store its audit data indicative of a potential security violation on an external server, and also support receipt of the audit data, over a trusted channel.

Management: IDS_RDR_EXT.1
a) No management activities foreseen.

Audit: IDS_RDR_EXT.1

a) No auditable events are foreseen.

IDS_RDR_EXT.1 Restricted data review
Hierarchical to: No other components
IDS_RDR_EXT.1.1

The TSF shall be able to [selection: transmit the generated audit data indicative of a potential security violation to an external IT entity over a trusted channel defined in FTP_ITC.1, receive and store audit data indicative of a potential security violation from an external IT entity over a trusted channel defined in FTP_ITC.1].

Dependencies: FAU_GEN.1 Audit data generation

5.1.6.4 Family IDS_SDC_EXT: System data collection

Family Behaviour

This family defines the requirements for recording the occurrence of intrusion detection events that take place under TSF control. This family identifies the level of system data collection, enumerates the types of events that shall be collected by the TSF, and identifies the minimum set of IDS-related information that should be provided within various IDS record types.

Component Leveling

Figure 22 – System data collection family decomposition

"IDS_SDC_EXT.1 System data collection" defines the level of IDS events, and specifies the list of data that shall be recorded in each record.

Management: IDS_SDC_EXT.1

a) There are no management activities foreseen.

Audit: IDS_SDC_EXT.1

a) There are no auditable events foreseen.

IDS_SDC_EXT.1 System data collection
Hierarchical to: No other component
IDS_SDC_EXT.1.1

The TSF shall be able to collect the following information from the targeted IT System resource(s):

a) [selection: Start-up and shutdown, identification and authentication events, data accesses, service requests, network traffic, security configuration changes, data introduction, detected malicious code, access control configuration, service configuration, authentication configuration, accountability policy configuration, detected known vulnerabilities]

b) [assignment: *other specifically defined events*].

IDS_SDC_EXT.1.2

At a minimum, the TSF shall collect and record the following information:

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

b) The additional information specified in the *Details* column of Table 9 below.

Table 9 – IDS_SDC_EXT.1 (Explicit SFR Definition) Collected Events

Component	Event	Details
IDS_SDC_EXT.1	Network traffic	Protocol, source address, destination address

Dependencies: **FPT_STM.1 Reliable time stamps**

5.2 Extended TOE Security Assurance Components

There are no extended TOE Security Assurance Components.

6 Security Requirements

This section defines the SFRs and SARs met by the TOE. These requirements are presented following the conventions identified in Section 6.1.

6.1 Conventions

There are several font variations used within this ST. Selected presentation choices are discussed here to aid the Security Target reader.

The CC allows for assignment, refinement, selection and iteration operations to be performed on security functional requirements. All of these operations are used within this ST. These operations are performed as described in Part 2 of the CC, and are shown as follows:

a) Completed assignment statements are identified using [*italicized text within brackets*].
b) Completed selection statements are identified using [underlined text within brackets].
c) Refinements are identified using **bold text**. Any text removed is stricken (Example: ~~TSF Data~~) and should be considered as a refinement.
d) Extended Functional and Assurance Requirements are identified using "_EXT" at the end of the short name.
e) Iterations are identified by appending a number in parentheses following the component title. For example, FAU_GEN.1(1) Audit Data Generation would be the first iteration and FAU_GEN.1(2) Audit Data Generation would be the second iteration.

6.2 Security Functional Requirements

This section specifies the SFRs for the TOE. This section organizes the SFRs by CC class. Table 10 identifies all SFRs implemented by the TOE and indicates the ST operations performed on each requirement.

Table 10 – TOE Security Functional Requirements

Name	Description	S	A	R	I
FAU_GEN.1	Audit Data Generation	✓	✓		
FAU_GEN.2	User Identity Association				
FAU_STG_EXT.1	External Audit Trail Storage	✓			
FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)		✓	✓	
FCS_CKM_EXT.4	Cryptographic Key Zeroization				
FCS_COMM_PROT_EXT.1	Communications Protection	✓			
FCS_COP.1(1)	Cryptographic Operation (for data encryption/decryption)		✓	✓	✓
FCS_COP.1(2)	Cryptographic Operation (for cryptographic signature)		✓	✓	✓
FCS_COP.1(3)	Cryptographic Operation (for cryptographic hashing)		✓	✓	✓
FCS_COP.1(4)	Cryptographic Operation (for keyed-hash message authentication)		✓	✓	✓

FCS_HTTPS_EXT.1	HTTPS				
FCS_IPSEC_EXT.1	IPsec	✓	✓		
FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)	✓			
FCS_SSH_EXT.1	SSH	✓	✓		
FCS_TLS_EXT.1	TLS	✓			
FDP_IFC.1	Subset information flow control		✓		
FDP_IFF.1	Simple security attributes		✓	✓	
FDP_RIP.2	Full residual information protection	✓		✓	
FIA_PMG_EXT.1	Password Management				
FIA_UAU.6	Re-authenticating		✓		
FIA_UAU.7	Protected Authentication Feedback		✓		
FIA_UAU_EXT.5	Password-based Authentication Mechanism	✓	✓		
FIA_UIA_EXT.1	User identification and authentication	✓			
FMT_MSA.1	Management of security attributes	✓	✓		
FMT_MSA.3	Static attribute initialisation	✓	✓		
FMT_MTD.1	Management of TSF data	✓	✓		
FMT_SMF.1	Specification of Management Functions		✓		
FMT_SMR.1	Security roles		✓		
FPT_PTD_EXT.1(1)	Management of TSF Data (for reading of authentication data)			✓	✓
FPT_PTD_EXT.1(2)	Management of TSF Data (for reading of all symmetric keys)			✓	✓
FPT_RPL.1	Replay Detection		✓		
FPT_STM.1	Reliable time stamps			✓	
FPT_TST_EXT.1	TSF Testing				
FPT_TUD_EXT.1	Trusted Update				
FRU_RSA.1	Maximum Quotas	✓	✓		
FTA_SSL.3	TSF-initiated Termination		✓		
FTA_SSL_EXT.1	TSF-initiated Session Locking	✓			
FTA_TAB.1	Default TOE Access Banners			✓	
FTP_ITC.1(1)	Inter-TSF Trusted Channel (prevention of disclosure)	✓	✓	✓	✓
FTP_ITC.1(2)	Inter-TSF Trusted Channel (detection of modification)	✓	✓	✓	✓
FTP_TRP.1(1)	Trusted Path (prevention of disclosure)	✓	✓	✓	✓
FTP_TRP.1(2)	Trusted Path (detection of modification)	✓	✓	✓	✓
IDS_ANL_EXT.1	Analyzer analysis	✓	✓		

IDS_RCT_EXT.1	Analyzer react		✓		
IDS_RDR_EXT.1	Restricted Data Review	✓			
IDS_SDC_EXT.1	System Data Collection	✓	✓		

Note: S=Selection; A=Assignment; R=Refinement; I=Iteration

6.2.1 Class FAU: Security Audit

FAU_GEN.1 Audit Data Generation
Hierarchical to: No other components.
FAU_GEN.1.1

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

a) Start-up and shutdown of the audit functions;
b) All auditable events, for the [*not specified*] level of audit; and
c) [*All administrative actions;*
d) *Specifically defined auditable events listed in Table 11 below*].

Table 11 – FAU_GEN.1 Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	
FAU_GEN.2	None.	
FAU_STG_EXT.1	None.	
FCS_CKM.1	Failure on invoking functionality.	No additional information.
FCS_CKM_EXT.4	Failure on invoking functionality.	No additional information.
FCS_COMM_PROT_EXT.1	None.	
FCS_COP.1(1)	Failure on invoking functionality.	No additional information.
FCS_COP.1(2)	Failure on invoking functionality.	No additional information.
FCS_COP.1(3)	Failure on invoking functionality.	No additional information.
FCS_COP.1(4)	Failure on invoking functionality.	No additional information.
FCS_HTTPS_EXT.1	None	
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA. Establishment/termination of an IPsec SA.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_RBG_EXT.1	Failure on the randomization process.	No additional information.
FCS_SSH_EXT.1	None	
FCS_TLS_EXT.1	None	
FDP_IFC.1	None.	
FDP_IFF.1	All requests for and decisions about information flows.	The presumed addresses of the source and destination subject.
FDP_RIP.2	None.	
FIA_PMG_EXT.1	None.	
FIA_UAU_EXT.5	None	
FIA_UAU.6	Attempt to re-authenticate.	Origin of the attempt (e.g., IP address).

Requirement	Auditable Events	Additional Audit Record Contents
FIA_UAU.7	None.	
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FMT_MSA.1	None.	
FMT_MSA.3	None.	
FMT_MTD.1	None.	
FMT_SMF.1	None.	
FMT_SMR.1	None.	
FPT_PTD_EXT.1(1)	None.	
FPT_PTD_EXT.1(2)	None.	
FPT_RPL.1	None	
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT_TST_EXT.1	Indication that TSF self-test was completed.	Any additional information generated by the tests beyond "success" or "failure".
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FRU_RSA.1	Maximum quota being exceeded.	Resource identifier.
FTA_SSL_EXT.1	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_TAB.1	None.	
FTP_ITC.1(1)	None	
FTP_ITC.1(2)	None	
FTP_TRP.1(1)	None	
FTP_TRP.1(2)	None	
IDS_ANL_EXT.1	None.	
IDS_RCT_EXT.1	None.	
IDS_RDR_EXT.1	None.	
IDS_SDC_EXT.1	None.	

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 ST, [*information specified in column three of Table 11 above*].

Dependencies: FPT_STM.1 Reliable time stamps

FAU_GEN.2 User identity association

Hierarchical to: No other components.

FAU_GEN.2.1

For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

Dependencies: FAU_GEN.1 Audit data generation
FIA_UID.1 Timing of identification

FAU_STG_EXT.1 External audit trail storage

Hierarchical to: No other components.

FAU_STG_EXT.1.1

The TSF shall be able to [transmit the generated audit data to an external IT entity over a trusted channel defined in FTP_ITC.1].

Dependencies: FAU_GEN.1 Audit data generation
FTP_ITC.1(1) Inter-TSF trusted channel (prevention of disclosure)
FTP_ITC.1(2) Inter-TSF trusted channel (prevention of modification)

6.2.2 Class FCS: Cryptographic Support

FCS_CKM.1 Cryptographic key generation

Hierarchical to: No other components.

FCS_CKM.1.1

The TSF shall generate **asymmetric** cryptographic keys in accordance with a **domain parameter generator and [a random number generator]** ~~specified cryptographic key generation algorithm [assignment: *cryptographic key generation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*]~~ that meet the following: [

- **ANSI[32] X9.31 (1998), "Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry (rDSA), 1998"**
- **Generated key strength shall be equivalent to, or greater than, a symmetric key strength of 112 bits using conservative estimates.**
- **NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography"**

].

Dependencies:

FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM_EXT.4 Cryptographic Key Zeroization

Hierarchical to: No other components

FCS_CKM_EXT.4.1

The TSF shall zeroize all plaintext secret and private cryptographic keys and CSP[33]s when no longer required.

Dependencies: FCS_CKM.1 Cryptographic key generation]

FCS_COMM_PROT_EXT.1 Communications Protection

Hierarchical to: No other components

FCS_COMM_PROT_EXT.1.1

The TSF shall protect communications using [IPsec, SSH] and [TLS/HTTPS].

Dependencies: [FCS_IPSEC_EXT.1 IPSEC or
FCS_SSH_EXT.1 SSH]
FCS_HTTPS_EXT.1 HTTPS, if selected
FCS_TLS_EXT.1 TLS.

FCS_COP.1(1) Cryptographic operation (for data encryption/decryption)

Hierarchical to: No other components.

FCS_COP.1.1(1)

The TSF shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm [*AES*[34] *operating in* ***[CBC[35], CTR[36] modes]***] and cryptographic key sizes [assignment: *cryptographic key sizes 128-bits, 256-bits, and* ***[192-bits]***] that meet the following: [***FIPS PUB 197 "Advanced Encryption Standard (AES)", NIST SP[37] 800-38A, NIST SP 800-38D***].

Dependencies:

FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

[32] ANSI: American National Standards Institute

[33] CSP: Critical Security Parameter(s)

[34] AES: Advanced Encryption Standard

[35] CBC: Cipher Block Chaining

[36] CTR: Counter

[37] SP: Special Publication

FCS_COP.1(2) Cryptographic operation (for cryptographic signature)

Hierarchical to: No other components.

FCS_COP.1.1(2)

The TSF shall perform [*cryptographic signature services*] in accordance with a **[Digital Signature Algorithm (DSA) with a key size (modulus) of 2048 bits or greater, RSA[38] Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater]** ~~specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*]~~ that meet the following: [**FIPS PUB 186-3, "Digital Signature Standard"**].

Dependencies: [

FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)

Hierarchical to: No other components.

FCS_COP.1.1(3)

The TSF shall perform [*cryptographic hashing services*] in accordance with a specified cryptographic algorithm [**SHA[39]-1, SHA_256, SHA-384, SHA-512] and message digest sizes [160, 256, 384, 512] bits** ~~and cryptographic key sizes [assignment: *cryptographic key sizes*]~~ that meet the following: [*FIPS Pub 180-3, "Secure Hash Standard."*].

Dependencies:

FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1(4) Cryptographic operation (for keyed-hash message authentication)

Hierarchical to: No other components.

FCS_COP.1.1(4)

The TSF shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm HMAC-[**SHA-1, SHA-256, SHA-384, SHA-512**], **key size [*160-bit, 256-bit, 384-bit, 512-bit*], and message digest sizes [160, 256, 384, 512] bits** ~~and cryptographic key sizes [assignment: *cryptographic key sizes*]~~ that meet the following: [*FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code, and FIPS Pub 180-3, "Secure Hash Standard."*].

Dependencies:

FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_HTTPS_EXT.1 HTTPS

Hierarchical to: No other components

FCS_HTTPS_EXT.1.1

The TSF shall implement the HTTPS protocol that complies with RFC 2818.

FCS_HTTPS_EXT.1.2

The TSF shall implement the HTTPS protocol using TLS as specified in FCS_TLS_EXT.1.

Dependencies: FCS_TLS_EXT.1 TLS

FCS_IPSEC_EXT.1 IPsec

Hierarchical to: No other components

FCS_IPSEC_EXT.1.1

The TSF shall implement IPsec using the ESP protocol as defined by RFC 4303 using the cryptographic algorithms AES-CBC-128, AES-CBC-256 (both specified by RFC 3602), [no other algorithms] and using IKEv1 as defined in RFCs 2407, 2408, 2409, and RFC 4109; [no other method] to establish the security association.

FCS_IPSEC_EXT.1.2

[38] RSA: Rivest Shamir Adleman

[39] SHA: Secure Hash Algorithm

The TSF shall ensure that IKEv1 Phase 1 exchanges use only main mode.

FCS_IPSEC_EXT.1.3

The TSF shall ensure that IKEv1 SA lifetimes are able to be limited to 24 hours for Phase 1 SAs and 8 hours for Phase 2 SAs.

FCS_IPSEC_EXT.1.4

The TSF shall ensure that IKEv1 SA lifetimes are able to be limited to [*200* MB of traffic for Phase 2 SAs.

FCS_IPSEC_EXT.1.5

The TSF shall ensure that all IKE protocols implement DH Groups 14 (2048-bit MODP), and [*1 (768-bit MODP), 2 (1024-bit MODP), 5 (1536-bit MODP)*].

FCS_IPSEC_EXT.1.6

The TSF shall ensure that all IKE protocols implement Peer Authentication using the [DSA, rDSA] algorithm.

FCS_IPSEC_EXT.1.7

The TSF shall support the use of pre-shared keys (as referenced in the RFCs) for use in authenticating its IPsec connections.

FCS_IPSEC_EXT.1.8

The TSF shall support the following:

1. Pre-shared keys shall be able to be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”);

2. Pre-shared keys of 22 characters and [*pre-shared key lengths of 8 to 31*].

Dependencies: FCS_COP.1(1) Cryptographic operation (for data encryption/decryption)
FCS_COP.1(2) Cryptographic operation (for cryptographic signature)

FCS_RBG_EXT.1 Random Bit Generation

Hierarchical to: No other components

FCS_RBG_EXT.1.1

The TSF shall perform all random bit generation (RBG) services in accordance with [FIPS PUB 140-2 Annex C: X9.31 Appendix 2.4 using AES] seeded by an entropy source that accumulated entropy from at least one independent TSF-hardware-based noise source.

FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded with a minimum of [*128 bits*]of entropy at least equal to the greatest bit length of the keys and authorization factors that it will generate.

Dependencies: None.

FCS_SSH_EXT.1 SSH

Hierarchical to: No other components

FCS_SSH_EXT.1.1

The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, and 4254.

FCS_SSH_EXT.1.2

The TSF shall ensure that the SSH connection be rekeyed after no more than 2^{28} packets have been transmitted using that key.

FCS_SSH_EXT.1.3

The TSF shall ensure that the SSH protocol implements a timeout period for authentication as defined in RFC 4252 of [*1800 (30 minutes)*], and provide a limit to the number of failed authentication attempts a client may perform in a single session to [*three consecutive failed attempts*] attempts.

FCS_SSH_EXT.1.4

The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, password-based.

FCS_SSH_EXT.1.5

The TSF shall ensure that, as described in RFC 4253, packets greater than [*262144*] bytes in an SSH transport connection are dropped.

FCS_SSH_EXT.1.6

The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms: AES-CBC-128, AES-CBC-256, [no other algorithms]**.**

FCS_SSH_EXT.1.7

The TSF shall ensure that the SSH transport implementation uses SSH_RSA and [No other public key algorithms] as its public key algorithm(s).

FCS_SSH_EXT.1.8

The TSF shall ensure that data integrity algorithms used in SSH transport connection is [hmac-sha1].

FCS_SSH_EXT.1.9

The TSF shall ensure that diffie-hellman-group14-sha1 is the only allowed key exchange method used for the SSH protocol.

Dependencies: FCS_COP.1(1) Cryptographic operation (for data encryption/decryption)
FCS_COP.1(4) Cyprtographic operation (for keyed-hash message authentication)
FCS_COP.1(2) Cryptographic operation (for cryptographic signature)

FCS_TLS_EXT.1 TLS

Hierarchical to: No other components

FCS_TLS_EXT.1.1

The TSF shall implement one or more of the following protocols [TLS 1.0 (RFC 2246)] supporting the following ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA

[*None*
].

Dependencies: FCS_COP.1(2) Cryptographic operation (for cryptographic signature)
FCS_COP.1(1) Cyprtographic operation (for data encryption/decryption)
FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)

6.2.3 Class FDP: User Data Protection

FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.

FDP_IFC.1.1

The TSF shall enforce the [*UNAUTHENTICATED SFP*] on [

- *Subjects: unauthenticated external IT entities that send and receive information through the TOE to one another*
- *Information: traffic sent through the TOE from one subject to another*
- *Operation: pass information*

].

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFF.1 Simple security attributes

Hierarchical to: No other components.

FDP_IFF.1.1

The TSF shall enforce the [*UNAUTHENTICATED SFP*] based on **at least** the following types of subject and information security attributes: [

- *Subject security attributes:*
  - *presumed address*
  - *other subject security attributes as configured by the administrators*
- *Information security attributes:*
  - *presumed address of source subject*
  - *presumed address of destination subject*
  - *transport layer protocol*
  - *interface on which traffic arrives and departs*

].

FDP_IFF.1.2

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

- *Subjects on an internal network can cause information to flow through the TOE to another connected network if:*
  - *all the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from all possible combinations of the values of the information flow security attributes, created by the authorized administrator;*
  - *the presumed address of the source subject, in the information, translates to an internal network address;*
  - *and the presumed address of the destination subject, in the information, translates to an address on the other connected network.*
- *Subjects on the external network can cause information to flow through the TOE to another connected network if:*
  - *all the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from all possible combinations of the values of the information flow security attributes, created by the authorized administrator;*
  - *the presumed address of the source subject, in the information, translates to an external network address;*
  - *and the presumed address of the destination subject, in the information, translates to an address on the other connected network.*

].

FDP_IFF.1.3

The TSF shall enforce the [*none*].

FDP_IFF.1.4

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

FDP_IFF.1.5

The TSF shall explicitly deny an information flow based on the following rules: [

- *The TOE shall reject requests for access or services where the information arrives on an external TOE interface, and the presumed address of the source subject is an internal IT entity on an internal network;*
- *The TOE shall reject requests for access or services where the information arrives on either an internal or external TOE interface, and the presumed address of the source subject is an external IT entity on a broadcast network;*
- *The TOE shall reject requests for access or services where the information arrives on either an internal or external TOE interface, and the presumed address of the source subject is an external IT entity on the loopback network.*

].

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

FDP_RIP.2 Full residual information protection

Hierarchical to: FDP_RIP.1 Subset residual information protection

FDP_RIP.2.1

The TSF shall ensure that any previous information content of a **network packet or cryptographic critical security parameter** ~~resource~~ is made unavailable upon the [deallocation of the resource from] all objects.

Dependencies: No dependencies

6.2.4 Class FIA: Identification and Authentication

FIA_PMG_EXT.1 Password Management

Hierarchical to: No other components

FIA_PMG_EXT.1.1

The TSF shall provide the following password management capabilities for administrative passwords:

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”);
2. Minimum password length shall settable by the Security Administrator, and support passwords of 8 characters or greater;
3. Passwords composition rules specifying the types and number of required characters that comprise the password shall be settable by the Security Administrator.
4. Passwords shall have a maximum lifetime, configurable by the Security Administrator.
5. New passwords must contain a minimum of 4 character changes from the previous password.

Dependencies: No dependencies.

FIA_UAU_EXT.5 Password-based Authentication Mechanism

Hierarchical to: No other components

FIA_UAU_EXT.5.1

The TSF shall provide a local password-based authentication mechanism, [*RADIUS*] to perform user authentication.

FIA_UAU_EXT.5.2

The TSF shall ensure that users with expired passwords are [required to create a new password after correctly entering the expired password].

Dependencies: No dependencies.

FIA_UAU.6 Re-authenticating

Hierarchical to: No other components.

FIA_UAU.6.1

The TSF shall re-authenticate the user under the conditions [*when the user changes their password*, *no other conditions*].

Dependencies: No dependencies

FIA_UAU.7 Protected authentication feedback

Hierarchical to: No other components.

FIA_UAU.7.1

The TSF shall provide only [*obscured feedback*] to the user while the authentication is in progress.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_UIA_EXT.1 User Identification and Authentication

Hierarchical to: FIA_UAU.1, FIA_UID.1

FIA_UIA_EXT.1.1

The TSF shall allow [no services] on behalf of the user to be performed before the user is identified and authenticated.

FIA_UIA_EXT.1.2

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

Dependencies: No dependencies.

6.2.5 Class FMT: Security Management

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

FMT_MSA.1.1

The TSF shall enforce the [*UNAUTHENTICATED SFP*] to restrict the ability to [[*perform all administrative tasks on*]] the security attributes [*all security attributes*] to [*authorized administrators*].

Dependencies:

FDP_IFC.1 Subset information flow control
FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_MSA.3 Static attribute initialisation

Hierarchical to: No other components.

FMT_MSA.3.1

The TSF shall enforce the [*UNAUTHENTICATED SFP*] to provide [restrictive] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2

The TSF shall allow the [*authorized administrators*] to specify alternative initial values to override the default values when an object or information is created.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MTD.1 Management of TSF data

Hierarchical to: No other components.

FMT_MTD.1.1

The TSF shall restrict the ability to [[*manage*]] the [*TSF data*] to [*the Security Administrators*].

Dependencies: FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions: [

- *Ability to configure the list of TOE services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1, respectively.*
- *Ability to configure the cryptographic functionality.*
- *Ability to update the TOE, and to verify the updates using the digital signature capability (FCS_COP.1(2)) and [no other functions].*

].

Dependencies: FCS_COP.1(2) Cryptographic operation (for cryptographic signature)

FMT_SMR.1 Security roles

Hierarchical to: No other components.

FMT_SMR.1.1

The TSF shall maintain the roles [*Security Administrator, other authorized administrative roles as defined by administrators*].

FMT_SMR.1.2

The TSF shall be able to associate users with roles.

Dependencies: FIA_UID.1 Timing of identification

6.2.6 Class FPT: Protection of the TSF

FPT_PTD_EXT.1(1) Management of TSF data (for reading of authentication data)
Hierarchical to: No other components
FPT_PTD_EXT.1.1

The TSF shall **prevent** reading of *the plaintext passwords.*

Dependencies: No dependencies.

FPT_PTD_EXT.1(2) Management of TSF data (for reading of all symmetric keys)
Hierarchical to: No other components
FPT_PTD_EXT.1.1

The TSF shall **prevent** reading of all *pre-shared keys, symmetric key, and private keys.*

Dependencies: No dependencies.

FPT_RPL.1 Replay detection
Hierarchical to: No other components.
FPT_RPL.1.1

The TSF shall detect replay for the following entities: [*network packets terminated at the TOE that are transmitted through the use of the TSF cryptographic services IPSec, TLS, SNMPv3, and SSH*].

FPT_RPL.1.2

The TSF shall perform [*reject the data*] when replay is detected.

Dependencies: No dependencies

FPT_STM.1 Reliable time stamps
Hierarchical to: No other components.
FPT_STM.1.1

The TSF shall be able to provide reliable time stamps **for its own use**.

Dependencies: No dependencies

FPT_TST_EXT.1 TSF testing
Hierarchical to: No other components
FPT_TST_EXT.1.1

The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

Dependencies: No dependencies.

FPT_ TUD_EXT.1 Trusted Update
Hierarchical to: No other components
FPT_TUD_EXT.1.1

The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.

FPT_TUD_EXT.1.2

The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.3

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

Dependencies: FCS_COP.1(2) Cryptographic operation (for cryptographic signature)

6.2.7 Class FRU: Resource Utilization

FRU_RSA.1 Maximum quotas

Hierarchical to: No other components.

FRU_RSA.1.1

The TSF shall enforce maximum quotas of the following resources: [*simultaneous administrative connections*, *no other resource*] that [subjects] can use [simultaneously].

Dependencies: No dependencies

6.2.8 Class FTA: TOE Access

FTA_SSL_EXT.1 TSF-initiated session locking

Hierarchical to: No other components

FTA_SSL_EXT.1.1

The TSF shall, for local interactive sessions, [terminate the session] after a Security Administrator-specified time period of inactivity.

Dependencies: FIA_UIA_EXT.1 User identification and authentication.

FTA_SSL.3 TSF-initiated termination

Hierarchical to: No other components.

FTA_SSL.3.1

The TSF shall terminate **a remote** ~~an~~ interactive session after a [*Security Administrator-configurable time interval of session inactivity*].

Dependencies: No dependencies

FTA_TAB.1 Default TOE access banners

Hierarchical to: No other components.

FTA_TAB.1.1

Before establishing a user/**administrator** session, the TSF shall display **a Security Administrator-specified** ~~an~~ advisory **notice and consent** warning message regarding unauthorised use of the TOE.

Dependencies: No dependencies

6.2.9 Class FTP: Trusted Path/Channels

FTP_ITC.1(1) Inter-TSF trusted channel (prevention of disclosure)

Hierarchical to: No other components.

FTP_ITC.1.1(1)

The TSF shall **use [*IPsec*] to** provide a **trusted** communication channel between itself and ~~another trusted IT product~~ **authorized IT entities** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from ~~modification or~~ disclosure.

FTP_ITC.1.2(1)

The TSF shall permit [the TSF**, or the authorized IT entities**] to initiate communication via the trusted channel.

FTP_ITC.1.3(1)

The TSF shall initiate communication via the trusted channel for [*all IPsec communications*].

Dependencies: No dependencies

FTP_ITC.1(2) Inter-TSF trusted channel (detection of modification)

Hierarchical to: No other components.

FTP_ITC.1.1(2)

The TSF shall **use [*IPsec*] in providing** ~~provide~~ a **trusted** communication channel between itself and **authorized IT entities** ~~another trusted IT product~~ that is logically distinct from other communication channels and provides assured identification of its end points and **detection of the modified data** ~~protection of the channel data from modification or disclosure~~.

FTP_ITC.1.2(2)

The TSF shall permit [the TSF**, or the authorized IT entities**] to initiate communication via the trusted channel.

FTP_ITC.1.3(2)

The TSF shall initiate communication via the trusted channel for [*all IPsec communications*].

Dependencies: No dependencies

FTP_TRP.1(1) Trusted path (prevention of disclosure)

Hierarchical to: No other components.

FTP_TRP.1.1(1)

The TSF shall provide a communication path between itself and [remote] **administrators** ~~users~~ **using [*HTTPS, SSH, and SNMP using AES as specified in FCS_COP.1(1) and SHA as specified in FCS_COP.1(3)*]** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [disclosure].

FTP_TRP.1.2(1)

The TSF shall permit [remote **administrators** ~~users~~] to initiate communication via the trusted path.

FTP_TRP.1.3(1)

The TSF shall require the use of the trusted path for [*all remote administrative actions*].

Dependencies: No dependencies

FTP_TRP.1(2) Trusted path (detection of modification)

Hierarchical to: No other components.

FTP_TRP.1.1(2)

The TSF shall provide a communication path between itself and [remote] **administrators** ~~users~~ **using [*HTTPS, SSH, and SNMP using AES as specified in FCS_COP.1(1) and SHA as specified in FCS_COP.1(3)*]]** that is logically distinct from other communication paths and provides assured identification of its end points and **detection of modification of the**

communicated data ~~protection of the communicated data from [selection: modification, disclosure, [*assignment: other types of integrity or confidentiality violation*]]~~.

FTP_TRP.1.2(2)

The TSF shall permit [remote **administrators** ~~users~~] to initiate communication via the trusted path.

FTP_TRP.1.3(2)

The TSF shall require the use of the trusted path for [*all remote administrative actions*].

Dependencies: No dependencies

6.2.10 Class IDS: Intrusion Detection Function

IDS_ANL_EXT.1 Analyzer analysis
Hierarchical to: No other components
IDS_ANL_EXT.1.1
The TSF shall perform the following analysis function(s) on all system data collected:

a) [statistical, signature]; and

b) [*no other analytical functions*].

IDS_ANL_EXT.1.2
The System shall record within each analytical result at least the following information:

a) Date and time of the result, type of result, identification of data source; and

b) [*no other security relevant information about the result*].

Dependencies: IDS_SDC_EXT.1 System Data Collection
FPT_STM.1 Reliable Timestamps

IDS_RCT_EXT.1 Analyzer react
Hierarchical to: No other components
IDS_RCT_EXT.1.1
The TSF shall send an alarm to [*the external syslog server*] and take [*no other action*] upon detection of a potential security violation.
Dependencies: IDS_SDC_EXT.1 System data collection

IDS_RDR_EXT.1 Restricted data review
Hierarchical to: No other components
IDS_RDR_EXT.1.1
The TSF shall be able to [transmit the generated audit data indicative of a potential security violation to an external IT entity over a trusted channel defined in FTP_ITC.1].
Dependencies: FAU_GEN.1 Audit data generation

IDS_SDC_EXT.1 System data collection
Hierarchical to: No other component
IDS_SDC_EXT.1.1
The TSF shall be able to collect the following information from the targeted IT System resource(s):

a) [network traffic]

b) [*no other specifically defined events*].

IDS_SDC_EXT.1.2
At a minimum, the TSF shall collect and record the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) The additional information specified in the *Details* column of Table 12 below.

Table 12 – IDS_SDC_EXT.1 Collected Events

Component	Event	Details

Component	Event	Details
IDS_SDC_EXT.1	Network traffic.	Protocol, source address, destination address.

Dependencies: **FPT_STM.1 Reliable time stamps**

6.3 Security Assurance Requirements

This section defines the assurance requirements for the TOE. Assurance requirements are taken from the CC Part 3 and are EAL 2 augmented with ALC_FLR.2. Table 13 summarizes the requirements.

Table 13 – EAL2 Assurance Requirements

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

7 TOE Summary Specification

This section presents information to detail how the TOE meets the functional requirements described in previous sections of this ST.

7.1 TOE Security Functions

Each of the security requirements and the associated descriptions correspond to the security functions. Hence, each function is described by how it specifically satisfies each of its related requirements. This serves to both describe the security functions and rationalize that the security functions satisfy the necessary requirements.

Table 14 – Mapping of TOE Security Functions to Security Functional Requirements

TOE Security Function	SFR ID	Description
Intrusion Detection	IDS_ANL_EXT.1	Analyzer analysis
	IDS_RCT_EXT.1	Analyzer react
	IDS_RDR_EXT.1	Restricted Data Review
	IDS_SDC_EXT.1	System Data Collection
Protected Communications	FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM_EXT.4	Cryptographic Key Zeroization
	FCS_COMM_PROT_EXT.1	Communications Protection
	FCS_COP.1(1)	Cryptographic Operation (for data encryption/decryption)
	FCS_COP.1(2)	Cryptographic Operation (for cryptographic signature)
	FCS_COP.1(3)	Cryptographic Operation (for cryptographic hashing)
	FCS_COP.1(4)	Cryptographic Operation (for keyed-hash message authentication)
	FCS_HTTPS_EXT.1	HTTPS
	FCS_IPSEC_EXT.1	IPsec
	FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
	FCS_SSH_EXT.1	SSH
	FCS_TLS_EXT.1	TLS
	FPT_PTD_EXT.1(2)	Management of TSF Data (for reading of all symmetric keys)
	FPT_RPL.1	Replay Detection
	FTP_ITC.1(1)	Inter-TSF Trusted Channel (prevention of disclosure)

TOE Security Function	SFR ID	Description
	FTP_ITC.1(2)	Inter-TSF Trusted Channel (detection of modification)
	FTP_TRP.1(1)	Trusted Path (prevention of disclosure)
	FTP_TRP.1(2)	Trusted Path (detection of modification)
Residual Information Clearing	FDP_RIP.2	Full residual information protection
Resource Availability	FRU_RSA.1	Maximum Quotas
System Monitoring	FAU_GEN.1	Audit Data Generation
	FAU_GEN.2	User Identity Association
	FAU_STG_EXT.1	External Audit Trail Storage
	FPT_STM.1	Reliable time stamps
TOE Administration	FIA_PMG_EXT.1	Password Management
	FIA_UAU.6	Re-authenticating
	FIA_UAU.7	Protected Authentication Feedback
	FIA_UAU_EXT.5	Password-based Authentication Mechanism
	FIA_UIA_EXT.1	User identification and authentication
	FMT_MSA.1	Management of security attributes
	FMT_MSA.3	Static attribute initialisation
	FMT_MTD.1	Management of TSF data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.1	Security roles
	FPT_PTD_EXT.1(1)	Management of TSF Data (for reading of all symmetric keys)
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_TAB.1	Default TOE Access Banners
Traffic Filter Firewall	FDP_IFC.1	Subset information flow control
	FDP_IFF.1	Simple security attributes
TSF Self Test	FPT_TST_EXT.1	TSF Testing
Verifiable Updates	FCS_COP.1(2)	Cryptographic Operation (for cryptographic signature)
	FCS_COP.1(3)	Cryptographic Operation (for

TOE Security Function	SFR ID	Description
		cryptographic hashing)
	FPT_TUD_EXT.1	Trusted Update

7.1.1 Intrusion Detection

The Intrusion Detection function enables the TOE to collect data about network traffic on monitored networks, analyze the collected data for potential statistical and signature-based security violations, automatically react to detected potential security violations, and to allow authorized administrators to review the collected data and analyses.

The TOE collects the following information about network traffic on all monitored networks:

- Date and time the network packet was observed
- Protocol-type of the packet
- Packet source address and destination address

The TOE constantly analyzes collected data for potential security violations based on the network traffic's attributes, signatures and statistical models provided and periodically updated by Curtiss-Wright, and arbitrary rules created and managed by authorized administrators. When network traffic is deemed by the TOE to represent a potential security violation, an alert is generated and logged for administrative review, and other actions can also be taken (as configured by the administrator), such as sending SNMP trap. The TOE protects the collected data from review by anyone except those administrators who have been granted permission to view it.

TOE Security Functional Requirements Satisfied: IDS_ANL.1, IDS_RCT.1, IDS_RDR.1, IDS_SDC.1.

7.1.2 Protected Communications

The Protected Communications TSF ensures that all types of communications in which the TOE participates are secure. These types of communications are:

1. The TOE communicating with a remote administrator;
2. The TOE communicating with another IT entity that is not another instance of the TOE.

Since plaintext communication with the TOE might allow critical data (such as passwords, configuration setting, or routing data) to be intercepted (disclosed), manipulated (modified), or replayed by intermediate systems, the TOE implements a FIPS 140-2 validated cryptographic module and uses it to encrypt (and, when appropriate, digitally sign) all such critical communications. Specifically, the HTTPS and SSHprotocols, and SNMPv3 secure mechanisms are used to protect administrative communication sessions. Each secure session is distinguished by session information, which is protected by the session protocol (HTTPS, SSH or SNMP). The administrator can access the module via the CLI or the Web Interface. The administrator uses secure pipes or tunnels with HTTPS, SSH, or SNMPv3 secure mechanisms. Each administrator must authenticate using the user ID and password or certificates associated with the correct protocol in order to set up the secure tunnel. The TOE then follows the appropriate protocol to distinguish between simultaneous administrators. Each session remains active (logged in) and secured using the tunneling protocol until the administrator logs out.

In order for Security Administrators to administer the TOE using HTTPS, , the secure HTTP server available on the TOE requires a certificate to be generated and installed prior to enabling the secure HTTP server. The web browser in which the remote administrator must authenticate with must accept the certificate offered by the secure HTTP server in order to achieve connection. Security Administrators are also required to provide proper identification and authentication in order to access the Web Interface.

The TOE implements SSH to protect communication between an administrator using the CLI and the TOE. The TOE rekeys an SSH connection before more than 2(28) packets have been sent with a given key. Configuration is not required to achieve this. The TOE drops an SSH session connection after a number of failed authentication attempts. The number is configurable through the CLI. The timeout period is thirty minutes. SSH utilizes 3DES-CBC, AES-CBC-128, and AES-CBC-256 for encryption. Password-based authentication methods are available and utilize HMAC-SHA1 as the authentication algorithm. Large packets are detected and handled by the OpenSSL module. The max packet size is 256KB and are handled according to RFC 4253. SSH management of the TOE is also capable of public-key authentication.

The TOE implements SNMPv3 to protect communications between a client and the TOE. SNMPv3 utilites encryption and decryptions functions in using AES in CBC and CTR modes. Cryptographic hashing services utilized for SNMPv3 communication between a client and the TOE uses the SHA algorithm in accordance to FIPS Pub 180-3, "*Secure Hashing".*

The TOE implements IPsec to protect communication between the TOE and another authorized IT entity. The TOE restricts the ability to configure "Confidentiality Only" ESP mode for IPsec which is disabled. Both certificate and pre-shared key method of IKE peer authentication is supported by the TOE. The TOE supports both manual key IPsec and IKE (v1/v2) based key negotiation for IPsec implementation. The pre-shared key in IKE is used for authenticating the peer in the Phase-1 exchange of the IKE negotiation.

The TOE implements DH Groups key exchange available in OpenSSL-0.9.8r according to RFC 3526. IKE DH group key exchange implementation is based on RFC 2409. The TOE implements DH Groups 14 (2048-bit MODP), 1 (768-bit MODP), 2 (1024-bit MODP), and 5 (1536-bit MODP). Groups can be selected through the management interface.

The TOE employs TLS in accordance to RFC 2246 and supports the following ciphersuites:

- TLS_RSA_WITH_AES[40]_128_CBC[41]_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA

The TOE enforces the Protected Communication TSF as encryption and decryption is provided using AES in CBC, ECB, CFB128, CTR, and CMAC modes using 128 bit and 256 bit keys in accordance to FIPS PUB 197, NIST SP 800-38A, and NIST SP800-38D. The TOE utilizes cryptographic signature services using DSA with a key size of 2048 bits to 4096 bits, and RSA PKCS#1 v1.5 Signature Generation/Verification with a key size of 2048 bits to 4096 bits in accordance to FIPS PUB 186-3 "Digital Signature Standard. Cryptographic hashing services are accomplished by using SHA-1, SHA-256, SHA-384, and SHA-512 cryptographic algorithms with key sizes of 160, 256, 384, 512 bits according to FIPS PUB 180-3 "Secure Hash Standard". Keyed-hash message authentication is performed using the same SHA algorithms and message digest sizes of 160, 256, 384, and 512 bits according to FIPS PUB 198-1 "Keyed-Hash Message Authentication Code" and FIPS PUB 180-3 "Secure Hash Standard". The TOE utilizes asymmetric cryptographic keys using ANSI X9.31 PRNG with the generated key strength of 112 bits.

The FIPS 140-2 validated cryptographic module ensures that all cryptographic operations are performed in a secure manner, and that all cryptographic critical security parameters (CSPs) are securely managed and zeroized when they are no longer needed. The TOE also prevents the reading of pre-shared keys, symmetric keys, and private keys in plaintext. Table 15 below provides details on how pre-shared keys, symmetric keys, and private keys are obscured to prevent reading.

The FIPS 140-2 validated cryptographic module implements a FIPS-approved random bit generator (RBG) and ensures that it generates random numbers in a FIPS-approved manner. The cryptographic module uses an ANSI x9.31 PRNG as defined in FIPS PUB 140-2 Annex C . Hardware-based noise sources for entropy

[40] AES – Advanced Encryption Standard

[41] CBC – Cipher Block Chaining

are retrieved from several places. In kernel mode, sources of randomness from the environment include inter-keyboard timings and inter-interrupt timings from network device drivers which are non-deterministic. Randomness from these sources is added to the entropy pool. In the user space, a combination of the PRNG, current system time (including of micro seconds), and process IDs are used as the sources of randomness. There are enough entropy sources for the entropy pool for the resulting RBG output to be completely independent from time and environmental conditions. When the module requires entropy it makes a call to pull 32 bits of entropy from the entropy pool mentioned above. In the event of an entropy source failure, such as failure to produce 32 bits of entropy, the cryptographic module that contains the PRNG will receive an error instead of the requested entropy bits. Upon receipt of this error the module will unload and not provide random numbers. The module is restarted the next time any TOE component calls for cryptographic services.

Once the entropy input has been independently collected from the environment, the entropy input is put through a series of transformation. The rand function of the cryptographic module is used by the TOE to generate a random bit. Using rand, HMAC-SHA2 is applied on the plain text generated and the key generated. The generated HMAC_SHA2 digest is used as the PRNG key for the AES 128-bit algorithm. HMAC_SHA2 is again applied on the plain text generated and the key generated. The second HMAC_SHA2 digest is used as the seed for the AES 128-bit algorithm. By applying AES 128-bit encryption based on the above generated PRNG key and seed, the random number is generated. A 128-bit seed value is used for this process.

Table 15 below provides the details about the keys, key components, and critical security parameters (CSPs) used by the TOE when it is operating in the CC-certified configuration. Note that the CC-certified configuration requires that the TOE also be operating in the FIPS-approved mode of operation (“FIPS-mode”). For details on FIPS-mode, see the TOE’s administrative guidance and related FIPS 140-2 Security Policy.

Table 15 – TOE Keys, Key Components, and CSPs

CSP/Key	Type	Input	Output	Storage	Zeroization	Use
PSK (Pre-shared key)	AES 256-bit key	Pre-installed at factory	Never	Plaintext in RAM[42] or EEPROM[43]	By command, power cycle, reboot Procedure: overwrite with zeros	Encrypt the KEK[44]
KEK (Key encryption key)	AES 256-bit key	Generated internally	Encrypted with PSK or KEK	Plaintext in RAM, SRAM[45], or EEPROM	By command, power cycle, reboot Procedure: overwrite with zeros	Decrypt the DEK[46]

[42] RAM – Random Access Memory
[43] EEPROM – Electrically Erasable Programmable Read-Only Memory
[44] KEK: Key-Encrypting Key
[45] SRAM – Static Random Access Memory
[46] DEK: Data-Encrypting Key

CSP/Key	Type	Input	Output	Storage	Zeroization	Use
DEK (Data encryption key)	AES 256-bit key	Encrypted with KEK or generated internally	Never	Plaintext in RAM, SRAM, or EEPROM	By command, power cycle, reboot Procedure: overwrite with zeros	Encrypt and decrypt the data on flash
HMAC[47] key	AES 256-bit key	Generated internally	Never	Plaintext in RAM	By command, power cycle, reboot Procedure: overwrite with zeros	Message Authentication with SHS[48]
Admin/User password	Password	Plaintext	Never	Plaintext in RAM, SRAM, or EEPROM	By command Procedure: overwrite with zeros	Login for module management
DRBG[49] seed	Random value	Generated internally	Never	Plaintext in RAM	By command, power cycle, reboot Procedure: overwrite with zeros	Seed input to ANSI X9.31 Appendix 2.4 using AES PRNG

In order to ensure that the integrity and usefulness of the audit log is not compromised, the TOE securely transmits the audit log to an external syslog server.

The TOE includes an anti-Replay feature which detects and prevents replay attacks. Packets tranisimitted using the IPSec, TLS, SNMPv3 and SSH protocols and are associated with the replay will be rejected by the TOE once the replay has been detected.

TOE Security Functional Requirements Satisfied: FCS_CKM.1, FCS_CKM_EXT.4, FCS_COMM_PROT_EXT.1, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3) , FCS_COP.1(4), FCS_HTTPS_EXT.1, FCS_IPSEC_EXT.1, FCS_RBG_EXT.1, FCS_SSH_EXT.1, FCS_TLS_EXT.1, FPT_PTD_EXT.1(2), FPT_RPL.1, FTP_ITC.1(1), FTP_ITC.1(2), FTP_TRP.1(1), FTP_TRP.1(2).

7.1.3 Residual Information Clearing

The Residual Information Clearing TSF ensures that data is not accidentally "leaked" into network packets or cryptographic CSPs by ensuring that any data object representing a network packet or CSP is destroyed when that data object is no longer needed. CSPs are zeroized by the FIPS 140-2 validated cryptographic module, and network packet objects are zeroized by code outside of the FIPS module, when these objects are deallocated (that is, when they have been fully processed and are no longer needed. The TOE ensures that both CSPs and network packet objects are completely overwritten with zeros before the objects are

[47] HMAC: Hashed Message Authentication Code
[48] SHS – Secure Hash Standard
[49] DRBG: Deterministic RBG

deallocated, ensuring that any attempt to reconstruct the content of the object after deallocation will result in reconstruction of the zeros, not the actual CSP or packet data.

TOE Security Functional Requirements Satisfied: FDP_RIP.2.

7.1.4 Resource Availability

The Resource Availability TSF ensures that the TOE's resources supporting the administrative interfaces are not exhausted (causing failure of the TOE) by enforcing maximum quotas on the number of simultaneous administrative connections. The maximum quotas per connection type are:

- SSH: 8 simultaneous connections
- HTTPS: 10 simultaneous connections

TOE Security Functional Requirements Satisfied: FRU_RSA.1.

7.1.5 System Monitoring

The System Monitoring TSF generates audit data, ensuring that sufficient information exists to allow Security Administrators to discover both intentional and unintentional problems with the configuration or operation of the TOE. Each audited event is associated with the specific local administrator or remote entity that caused the event, and each event also has a reliable time stamp provided by the TOE. The TOE audits many events and system operations (documented in FAU_GEN.1), including (but not limited to):

- startup and shutdown of the audit functions
- all administrative actions
- all use of the identification and authentication functions
- failures of any TOE functions
- requests for information flows mediated by the TOE

The audit records are temporarily stored locally in a configurable number of rotating log files. Periodically, the TOE transmits the recorded audit data to an external syslog server for permanent storage and for review by the authorized administrator.

TOE Security Functional Requirements Satisfied: FAU_GEN.1, FAU_GEN.2, FAU_STG_EXT.1, FPT_STM.1.

7.1.6 TOE Administration

The TOE Administration TSF provides a trusted means for administrators to interact with the TOE for management purposes. The UNAUTHENTICATED SFP[50] is enforced by the TOE to ensure that only authorized administrators are allowed to perform administrative and management tasks on the TOE, and that restrictive default values are used for all security attributes used to enforce the UNAUTHENTICATED SFP.

Administrators can manage the TOE via either a secure web GUI secured via the HTTPS protocol, via a secure CLI protected via the SSH protocol, or via SNMP v3 protocol. CLI access via the serial port is disabled in FIPS-mode. The TOE can be configured by administrators to authenticate users either against a local password-based authentication mechanism or against a remote RADIUS[51] authentication server. The TSF requires that administrators use strong passwords that must be changed on a regular basis, and it requires users to re-authenticate when they change their passwords. Passwords are obscured during entry, to prevent "shoulder surfing" of administrative passwords. Passwords can be configured to expire, and

[50] SFP: Security Functional Policy

[51] RADIUS: Remote Authentication Dial In User Service

users are required to set a new password after correctly authenticating with an expired (and as-yet unchanged) password. The TOE does not provide the ability to view stored passwords. The TOE also prevents any user from reading any stored passwords by encrypted them. Passwords are stored as MD5 or SHA hash in RAM, SRAM, or EEPROM.

Unattended local or remote sessions are terminated after a configurable period of inactivity. A banner displaying configurable warning text is displayed to all users upon successful log in but before any actions can be taken. No services are available to users before they are successfully identified and authenticated, and only the services for which a user is authorized are provided to that user.

The TOE implements 15 privilege levels, each of which provides access to more commands than the previous (privilege level 1 provides very basic commands, and level 15 provides access to all commands). Level 15 privileges are limited to the TOE's root account and are disabled in the CC-evaluated configuration once the TOE has been set up. After initial set up of the TOE, the Security Administrator role referred to in this ST is assumed to have privilege level 14 or 15. The privilege level of each command can be customized by the Security Administrator.
TOE Security Functional Requirements Satisfied: FIA_PMG_EXT.1, FIA_UAU.6, FIA_UAU.7, FIA_UAU_EXT.5, FIA_UIA_EXT.1, FMT_MSA.1, FMT_MSA.3, FMT_MTD.1, FMT_SMF.1, FMT_SMR.1, FPT_PTD_EXT.1(1), FPT_PTD_EXT.1(2), FTA_SSL.3, FTA_SSL_EXT.1, FTA_TAB.1.

7.1.7 Traffic Filter Firewall

The Traffic Filter Firewall TSF defines and enforces the UNAUTHENTICATED information flow control Security Functional Policy (SFP). The UNAUTHENTICATED SFP ensures that the TOE mediates all attempts by unauthenticated external IT entities to send and receive data through the TOE to each other. The TOE determines whether or not to allow an information flow based on each external IT entity's presumed address and any other relevant security attributes as defined by the administrators in the SFP. It also analyzes the data to determine whether its presumed source address, presumed destination address, protocol, and the interface on which it arrives and departs matches the SFP rules. If the SFP rules allow the information flow to occur, then the data is passed out of the TOE on the appropriate interface; otherwise, the data is discarded and a record of the event is logged.

TOE Security Functional Requirements Satisfied: FDP_IFC.1, FDP_IFF.1.

7.1.8 TSF Self Test

The TSF Self Test TSF ensures that the TOE verifies the correct operation of critical TOE functions at power on and conditionally during TOE operation. The TOE performs the following self tests:

At power on:

1. Software Integrity Test: The TOE checks the integrity of its software using SHA1. The pre-boot secure firmware integrity is located in the hardware protect and is not modifiable. At power up the TOE computes a new digest and compares it to the pre computed digest value for the pre-boot secure image. This Integrity Test is also considered the KAT for SHA1. The TOE then continues to check the integrity of the loadable images using SHA1 and compares to the pre-computed digest for the loadable images. If the value are the same then the test passes, otherwise it fails.
2. Cryptographic Algorithm Tests:
    a. AES Known Answer Test (KAT): The AES KAT encrypts a known plaintext value with known keys. It then compares the resultant ciphertext with the expected ciphertext hard-coded in the TOE. If the two values differ, then the KAT fails. If the two values agree, the AES KAT then decrypts the ciphertext with the known keys and compares the decrypted text with the known plaintext. If they differ then the test fails. If they are the same, then the test passes.

b. SHA-256 KAT: The hashing algorithm performs a KAT for SHA-256. The KAT takes a specific value and hashes it. This digest value is then compared to the known value. If the values differ, the test fails. If they are the same, the test passes.
c. HMAC-SHA-256 KAT: The HMAC algorithm takes a known value and hashes it with a hard-coded HMAC key. The result is then compared to the expected value hard-coded in the module. If the values differ then the test fails. If they are the same, the test passes.
d. RBG KAT: A known seed value is used to initialize the RBG. A block of random data is then generated and compared to a pre-generated value. If these values are the same, the test passes. Otherwise, the test fails.

Conditionally during normal operation:

1. Continuous RBG Test: When a new random number is generated, it is first compared to the previously generated random number block. If they are equal then the test fails. If they differ, then the test passes, and the new random number is passed to the caller and stored in order to be compared to the next random number block.

If any of these tests fail, then the TOE will enter a critical error state, the TOE's Fault LED[52] will illuminate, and the TOE will require the administrator to clear the error condition by rebooting the TOE. While the TOE is in the error state, data output and cryptographic services are inhibited for untrusted interfaces until the error condition is cleared. Since these tests cover all of the TOE's critical functions, and since they are performed both at power-up and periodically during normal operation, and since they ensure that the TOE will enter a critical error state in which data output is inhibited if any of the tests fail, these tests are sufficient to demonstrate that the TSF is operating correctly at any point in time.

TOE Security Functional Requirements Satisfied: FPT_TST_EXT.1.

7.1.9 Verifiable Updates

The Verifiable Updates TSF enables the administrator to ensure that software or firmware updates are unmodified and are authentic before installation. Software and firmware updates are cryptographically hashed and signed FIPS 140-2 approved hashing and signing algorithms, and verified by the TOE's FIPS 140-2 validated cryptographic module. Administrators can retrieve the published cryptographic hash for the update from Curtiss-Wright and then verify the update's hash via the TOE before installation to be assured that the update was not corrupted or modified during storage or transit.

In order to initiate the update process, the administrator determines which of several update methods he wishes to use. These methods are detailed in the *VPX3-685 Secure Ethernet Router User's Manual*, Chapter 6, and generally involve powering off the TOE, physically installing several jumpers on the TOE's main board, and then booting the TOE into the desired update mode of operation. When the administrator supplies the TOE with the desired firmware update file, the TOE checks the digital signature on the update file and will only install the update if the signature is from Curtiss-Wright. The special portion of the TOE firmware that checks the signature and conducts the update cannot be updated by the administrator; the TOE must be returned to Curtiss-Wright for maintenance if that portion of the TOE firmware must be updated. All of this ensures that only valid, uncompromised, official updates from Curtiss-Wright can be installed on the TOE, and only by the authorized administrators (who have physical access to the TOE in order to install the jumpers).

TOE Security Functional Requirements Satisfied: FCS_COP.1(2), FCS_COP.1(3), FPT_TUD_EXT.1.

[52] LED: Light Emitting Diode

8 Rationale

8.1 Conformance Claims Rationale

This Security Target conforms to Part 2 and Part 3 of the *Common Criteria for Information Technology Security Evaluation*, Version 3.1 Revision 3. There are no protection profile conformance claims for this Security Target.

8.2 Security Objectives Rationale

This section provides a rationale for the existence of each threat, policy statement, and assumption that compose the Security Target. Sections 8.2.1, 8.2.2, and 8.2.3 demonstrate the mappings between the threats, policies, and assumptions to the security objectives are complete. The following discussion provides detailed evidence of coverage for each threat, policy, and assumption.

8.2.1 Security Objectives Rationale Relating to Threats

Table 16 – Threats:Objectives Mapping

Threats	Objectives	Rationale
T.ADMIN_ERROR An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.	O.TSF_SELF_TEST The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.	The TOE tests its security functionality to ensure that it is operating properly. The TOE also ensures that the FIPS 140-2 approved mode of operation is enforced properly.
T.ASPOOF An unauthorized person may carry out spoofing in which information flows through the TOE into a connected network by using a spoofed source address.	O.MEDIAT The TOE must mediate the flow of all information from users on a connected network to users on another connected network.	The TOE mediates all information flows from and to the protected network, allowing the TOE to allow or deny the flow if spoofing is detected.
T.FALACT An attacker might introduce identified or suspected vulnerabilities or perform inappropriate activity to which the TOE fails to react.	O.RESPON The TOE must respond appropriately to analytical conclusions.	The TOE reacts to analytical conclusions about suspected vulnerabilities or inappropriate activity.
T.FALASC An attacker might introduce vulnerabilities or perform inappropriate activity which the TOE fails to identify based on association of IDS data received from all data sources.	O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).	The TOE will recognize vulnerabilities or inappropriate activity from multiple data sources.

T.FALREC An attacker might introduce vulnerabilities or perform inappropriate activity which the TOE fails to recognize based on IDS data received from each data source.	O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).	The TOE will recognize vulnerabilities or inappropriate activity from a data source.
T.INADVE Inadvertent activity and access by attackers who are not TOE users may occur on an IT System the TOE monitors.	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The TOE collects audit and Sensor data.
T.MEDIAT An unauthorized person may send impermissible information through the TOE which results in the exploitation of resources on the internal network.	O.MEDIAT The TOE must mediate the flow of all information from users on a connected network to users on another connected network.	The TOE mediates all information that flows to and from the protected network.
T.MISACT Malicious activity by attackers who are not TOE users, such as introductions of Trojan horses and viruses, may occur on an IT System the TOE monitors.	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The TOE collects audit and Sensor data.
T.MISUSE Unauthorized accesses and activity (by attackers who are not TOE users) indicative of misuse may occur on an IT System the TOE monitors.	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The TOE collects audit and Sensor data.
T.RESOURCE_EXHAUSTION A process initiated by a TOE user, or a TOE user may deny access to TOE services by exhausting critical resources on the TOE.	O.RESOURCE_AVAILABILITY The TOE shall provide mechanisms that mitigate user attempts to exhaust TOE resources (e.g., persistent storage).	The TOE mitigates attempts (intentional or unintentional) to exhaust critical TOE resources.

T.SCNCFG Improper security configuration settings created by non-TOE users may exist in the IT System the TOE monitors.	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The TOE collects and stores static configuration information that might be indicative of a configuration setting change.
T.SCNMLC Users could execute malicious code on an IT System that the TOE monitors which causes modification of the IT System protected data or undermines the IT System security functions.	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The TOE collects and stores static configuration information that might be indicative of the presence of malicious code.
T.SCNVUL Vulnerabilities (introduced by non-TOE users) may exist in the IT System the TOE monitors.	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The TOE collects and stores static configuration information that might be indicative of the presence of a vulnerability.
T.TSF_FAILURE Security mechanisms of the TOE may fail, leading to a compromise of the TSF.	O.TSF_SELF_TEST The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.	The TOE tests its security functionality to ensure that it is operating properly. The TOE also ensures that the FIPS 140-2 approved mode of operation is enforced properly.
T.UNAUTHORIZED_ACCESS A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.	O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	The TOE displays a banner informing the user of the consequences of misusing the TOE.
	O.PROTECTED_COMMUNICATIONS The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and	The TOE protects communications channels to prevent malicious or accidental disclosure, hijacking, replay, or other compromise of authorized communications.
	O.SECSTA Upon initial start-up of the TOE or recovery from an interruption in TOE service, the TOE must not compromise its resources or those of any connected network.	The TOE ensures that no information is compromised by the TOE upon start-up or recovery.

	O.SESSION_LOCK The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked	The TOE terminates idle or unattended administrator sessions.
	O.SYSTEM_MONITORING The TOE will provide the capability to generate audit data and send those data to an external IT entity.	The TOE audits all administrative and authentication activities.
	O.TOE_ADMINISTRATION The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.	The TOE ensures that administrators (and only administrators) may administer the TOE.
T.UNAUTHORIZED_UPDATE A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.	O.VERIFIABLE_UPDATES The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.	The TOE allows the administrator to verify the integrity and authenticity of updates prior to installing them.
T.UNDETECTED_ACTIONS Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.	O.SYSTEM_MONITORING The TOE will provide the capability to generate audit data and send those data to an external IT entity.	The TOE audits all administrative and authentication activities.
	O.TOE_ADMINISTRATION The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.	The TOE ensures that administrators (and only administrators) may administer the TOE.
T.USER_DATA_REUSE User data may be inadvertently sent to a destination not intended by the original sender.	O.RESIDUAL_INFORMATION_ CLEARING The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.	The TOE ensures that any data contained in a protected resource is not reused with the resource is reallocated.

Every Threat is mapped to one or more Objectives in the table above. This complete mapping demonstrates that the defined security objectives counter all defined threats.

8.2.2 Security Objectives Rationale Relating to Policies

Table 17 – Policies:Objectives Mapping

Policies	Objectives	Rationale
P.ACCESS_BANNER The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.	O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	The TOE displays an appropriate advisory warning.

Every policy is mapped to one or more Objectives in the table above. This complete mapping demonstrates that the defined security objectives enforce all defined policies.

8.2.3 Security Objectives Rationale Relating to Assumptions

Table 18 – Assumptions:Objectives Mapping

Assumptions	Objectives	Rationale
A.NO_GENERAL_PURPOSE It is assumed that there are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.	OE.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.	No general-purpose computing capabilities are available on the TOE (beyond those required for TOE operation, administration, and support).
A.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.	OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.	The environment provides physical security for the TOE commensurate with the value of the TOE and its data.
A.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.	OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.	TOE Administrators are trusted to follow all administrative guidance.
A.TRUSTED NETWOK_BOUNDARY The TOE router controls the single access point to the trusted network and that there are no hostile entities on the trusted network side	OE.TRUSTED NETWOK_BOUNDARY The TOE router controls the single access point to the trusted network and that there are no hostile entities on the trusted network side	TOE controls the single entry point to the trusted network. Entities on the trusted network are not hostile.

Every assumption is mapped to one or more Objectives in the table above. This complete mapping demonstrates that the defined security objectives uphold all defined assumptions.

8.3 Rationale for Extended Security Functional Requirements

The extended requirements are defined in section 5. These SFRs exhibit functionality that can be easily documented in the ADV assurance evidence and thus do not require any additional Assurance Documentation.

8.4 Rationale for Extended TOE Security Assurance Requirements

There are no Extended SARs defined for this ST.

8.5 Security Requirements Rationale

The following discussion provides detailed evidence of coverage for each security objective.

8.5.1 Rationale for Security Functional Requirements of the TOE Objectives

Table 19 – Objectives:SFRs Mapping

Objective	Requirements Addressing the Objective	Rationale
O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	FTA_TAB.1 Default TOE Access Banners	The TOE displays an advisory notice and consent warning (specified by the administrator) before allowing an administrator to log in.
O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).	IDS_ANL_EXT.1 Analyzer analysis	The TOE accepts data from IDS Sensors or IDS Scanners and then applies analytical processes and information to derive conclusions about intrusions (past, present, or future).
O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	IDS_SDC_EXT.1 System Data Collection	The TOE collects and stores static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.

O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	IDS_SDC_EXT.1 System Data Collection	The TOE collects and stores information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the TOE.
O.MEDIAT The TOE must mediate the flow of all information from users on a connected network to users on another connected network.	FDP_IFC.1 Subset information flow control	The TOE controls the flow of all data covered by the UNAUTHENTICATED information flow control SFP.
	FDP_IFF.1 Simple security attributes	The TOE controls the flow of all data covered by the UNAUTHENTICATED information flow control SFP.
	FDP_RIP.2 Full residual information protection	The TOE does not use information that had previously flowed through the TOE, nor any TOE internal data, to pad packets passed through the TOE as part of an information flow.
	FMT_MSA.3 Static attribute initialisation	The TOE implements a default "deny" policy for information flow control security rules.
O.PROTECTED_COMMUNICATIONS The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and	FCS_CKM.1 Cryptographic Key Generation (for asymmetric keys)	The TOE uses only approved asymmetric key generation algorithms.
	FCS_CKM_EXT.4 Cryptographic Key Zeroization	The TOE zeroizes all plaintext secret and private keys and FIPS 140-2 CSPs when they are no longer needed.
	FCS_COMM_PROT_EXT.1 Communications Protection	The TOE uses only approved communication encryption protocols.
	FCS_COP.1(1) Cryptographic Operation (for data encryption/decryption)	The TOE uses only approved algorithms for data encryption and decryption.
	FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)	The TOE uses only approved algorithms for cryptographic signatures.
	FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)	The TOE uses only approved algorithms for cryptographic hashes.
	FCS_COP.1(4) Cryptographic Operation (for keyed-hash message authentication)	The TOE uses only approved algorithms for HMACs.

	FCS_HTTPS_EXT.1 HTTPS	The TOE uses only standards-compliant implementations of the HTTPS and TLS protocols.
	FCS_IPSEC_EXT.1 IPsec	The TOE uses only standards-compliant implementations of the Ipsec protocol, and that the Ipsec protocol is used in a secure manner.
	FCS_RBG_EXT.1 Cryptographic Operation (Random Bit Generation)	The TOE uses only standards-compliant implementations of the random bit generators, and that they are used in a secure manner.
	FCS_SSH_EXT.1 SSH	The TOE uses only standards-compliant implementations of the SSH protocol, and that the SSH protocol is used in a secure manner.
	FCS_TLS_EXT.1 TLS	The TOE uses only standards-compliant implementations of the TLS protocol, and that the TLS protocol is used in a secure manner.
	FPT_PTD_EXT.1(2) Management of TSF Data (for reading of all symmetric keys)	The TOE does not provide any interface or command which would allow an administrator to view plaintext pre-shared, symmetric, and private keys.
	FPT_RPL.1 Replay Detection	The TOE detects and rejects replayed network packets.
	FTP_ITC.1(1) Inter-TSF Trusted Channel (prevention of disclosure)	The TOE uses only approved communication protection algorithms to conduct communications with external authorized IT entities.
	FTP_ITC.1(2) Inter-TSF Trusted Channel (detection of modification)	The TOE uses only approved communication protection algorithms to conduct communications with external authorized IT entities.
	FTP_TRP.1(1) Trusted Path (prevention of disclosure)	The TOE uses only approved communication protection algorithms to conduct communications with remote administrators, preventing disclosure of the administrator's session data.

	FTP_TRP.1(2) Trusted Path (detection of modification)	The TOE uses only approved communication protection algorithms to conduct communications with remote administrators, allowing detection of any modification of the administrator's session data.
O.RESIDUAL_INFORMATION_CLEARING The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.	FDP_RIP.2 Full residual information protection	The TOE makes unavailable all previous information content of a resource when it is no longer in use.
O.RESOURCE_AVAILABILITY The TOE shall provide mechanisms that mitigate user attempts to exhaust TOE resources (e.g., persistent storage).	FRU_RSA.1 Maximum Quotas	The TOE enforces maximum quota usage of critical TOE resources, ensuring that those resources will not be exhausted.
O.RESPON The TOE must respond appropriately to analytical conclusions.	IDS_RCT_EXT.1 Analyzer react	The TOE responds appropriately to IDS-related analytical conclusions.
O.SECSTA Upon initial start-up of the TOE or recovery from an interruption in TOE service, the TOE must not compromise its resources or those of any connected network.	FMT_MSA.3 Static attribute initialisation	The TOE implements a default "deny" policy for information flow control security rules.
O.SESSION_LOCK The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked	FTA_SSL.3 TSF-initiated Termination	The TOE terminates unattended or idle administrative sessions after an administrator-specified period of time.
	FTA_SSL_EXT.1 TSF-initiated Session Locking	The TOE terminates unattended or idle administrative sessions after an administrator-specified period of time.
O.SYSTEM_MONITORING The TOE will provide the capability to generate audit data and send those data to an external IT entity.	FAU_GEN.1 Audit Data Generation	The TOE generates audit records of appropriate events.
	FAU_GEN.2 User Identity Association	The TOE associates actions of an identified user with that user.
	FAU_STG_EXT.1 External Audit Trail Storage	The TOE sends all audit records to the external audit log server.
	FPT_STM.1 Reliable time stamps	The TOE provides reliable time stamps for generated audit records.
O.TOE_ADMINISTRATION The TOE will provide mechanisms to ensure that only administrators	FIA_PMG_EXT.1 Password Management	The TOE requires secure passwords for administrative users.

are able to log in and configure the TOE, and provide protections for logged-in administrators.	FIA_UAU.6 Re-authenticating	The TOE requires administrators to re-authenticate when they perform actions that might indicate the presence of a different human operator using operating under the same credentials.
	FIA_UAU.7 Protected Authentication Feedback	The TOE does not display the user's password as it is being typed at the authentication prompt.
	FIA_UAU_EXT.5 Password-based Authentication Mechanism	The TOE supports the listed password-based authentication mechanisms.
	FIA_UIA_EXT.1 User identification and authentication	The TOE allows no services to be performed on behalf of the user before he is identified and authenticated.
	FMT_MSA.1 Management of security attributes	The TOE allows only authorized administrators the ability to perform administrative tasks on security attributes.
	FMT_MTD.1 Management of TSF data	The TOE allows only administrators to manage TSF data.
	FMT_SMF.1 Specification of Management Functions	The TOE provides administrators with the management functions required to perform their duties.
	FMT_SMR.1 Security roles	The TOE provides the administrative roles and privileges required to enable administrators to perform their duties.
	FPT_PTD_EXT.1(1) Management of TSF Data (for reading of authentication data)	The TOE does not provide any interface or command which would allow an administrator to view users' plaintext passwords.
	IDS_RDR_EXT.1 Restricted Data Review	The TOE restricts the review of IDS data to those granted explicit read-access.
O.TSF_SELF_TEST The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.	FPT_TST_EXT.1 TSF Testing	The TOE executes self-tests at power-up to ensure its correct operation.
O.VERIFIABLE_UPDATES The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the	FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)	The TOE ensures that updates are signed only with approved cryptographic signature algorithms.

administrator to be unaltered and (optionally) from a trusted source.	FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)	The TOE ensures that updates are hashed only with approved cryptographic hash algorithms.
	FPT_TUD_EXT.1 Trusted Update	The TOE administrators can verify the installed version of the TOE software/firmware, and can verify the integrity and authenticity of software/firmware updates prior to installing them.

8.5.2 Security Assurance Requirements Rationale

EAL2 was chosen used by the ST authors in order to provide a low to moderate level of assurance that is consistent with good commercial practices. Minimal additional tasks are placed upon the vendor assuming the vendor follows reasonable software engineering practices and can provide support to the evaluation for design and testing efforts. The chosen assurance level is appropriate with the threats defined for the environment. At EAL2, the TOE will have incurred a search for obvious flaws to support its introduction into the non-hostile environment. The augmentation of ALC_FLR.2 was chosen to give greater assurance of the developer's on-going flaw remediation processes.

8.5.3 Dependency Rationale

The SFRs in this ST satisfy all of the required dependencies listed in the Common Criteria and SFRs explicitly stated in this ST. Table 20 lists each requirement to which the TOE claims conformance and indicates whether the dependent requirements are included. As the table indicates, all dependencies have been met.

Table 20 – Functional Requirements Dependencies

SFR ID	Dependencies	Dependency Met	Rationale
FAU_GEN.1	FPT_STM.1	Met	
FAU_GEN.2	FAU_GEN.1	Met	
	FIA_UID.1	Met	Met by FIA_UIA_EXT.1, which is hierarchical to FIA_UID.1.
FAU_STG_EXT.1	FAU_GEN.1	Met	
	FTP_ITC.1(1)	Met	
	FTP_ITC.1(2)	Met	
FCS_CKM.1	FCS_CKM.4	Met	Met by the explicitly-stated FCS_CKM_EXT.4 requirement.
	FCS_COP.1	Met	Multiple iterations of FCS_COP.1 are included, all of which meet this dependency.
FCS_CKM_EXT.4	FCS_CKM.1	Met	

FCS_COMM_PROT_EXT.1	FCS_SSH_EXT.1	Met	
	FCS_HTTPS_EXT.1	Met	
	FCS_IPSEC_EXT.1	Met	
	FCS_TLS_EXT.1	Met	
FCS_COP.1(1)	FCS_CKM.1	Met	
	FCS_CKM.4	Met	Met by the explicitly-stated FCS_CKM_EXT.4 requirement.
FCS_COP.1(2)	FCS_CKM.1	Met	
	FCS_CKM.4	Met	Met by the explicitly-stated FCS_CKM_EXT.4 requirement.
FCS_COP.1(3)	FCS_CKM.1	Met	
	FCS_CKM.4	Met	Met by the explicitly-stated FCS_CKM_EXT.4 requirement.
FCS_COP.1(4)	FCS_CKM.1	Met	
	FCS_CKM.4	Met	Met by the explicitly-stated FCS_CKM_EXT.4 requirement.
FCS_HTTPS_EXT.1	FCS_TLS_EXT.1	Met	
FCS_IPSEC_EXT.1	FCS_COP.1(2)	Met	
	FCS_COP.1(1)	Met	
FCS_RBG_EXT.1	None	Met	
FCS_SSH_EXT.1	FCS_COP.1(1)	Met	
	FCS_COP.1(4)	Met	
	FCS_COP.1(2)	Met	
FCS_TLS_EXT.1	FCS_COP.1(2)	Met	
	FCS_COP.1(1)	Met	
	FCS_COP.1(3)	Met	
FDP_IFC.1	FDP_IFF.1	Met	
FDP_IFF.1	FDP_IFC.1	Met	
	FMT_MSA.3	Met	
FDP_RIP.2	None	Met	
FIA_PMG_EXT.1	None	Met	
FIA_UAU.6	None	Met	
FIA_UAU.7	FIA_UAU.1	Met	Met by FIA_UIA_EXT.1, which is hierarchical to FIA_UAU.1.

FIA_UAU_EXT.5	None	Met	
FIA_UIA_EXT.1	None	Met	
FMT_MSA.1	FMT_SMR.1	Met	
	FDP_IFC.1	Met	
	FMT_SMF.1	Met	
FMT_MSA.3	FMT_MSA.1	Met	
	FMT_SMR.1	Met	
FMT_MTD.1	FMT_SMF.1	Met	
	FMT_SMR.1	Met	
FMT_SMF.1	FCS_COP.1(2)	Met	
FMT_SMR.1	FIA_UID.1	Met	Met by FIA_UIA_EXT.1, which is hierarchical to FIA_UID.1.
FPT_PTD_EXT.1(1)	None	Met	
FPT_PTD_EXT.1(2)	None	Met	
FPT_RPL.1	None	Met	
FPT_STM.1	None	Met	
FPT_TST_EXT.1	None	Met	
FPT_TUD_EXT.1	FCS_COP.1(2)	Met	
FRU_RSA.1	None	Met	
FTA_SSL.3	None	Met	
FTA_SSL_EXT.1	FIA_UIA_EXT.1	Met	
FTA_TAB.1	None	Met	
FTP_ITC.1(1)	None	Met	
FTP_ITC.1(2)	None	Met	
FTP_TRP.1(1)	None	Met	
FTP_TRP.1(2)	None	Met	
IDS_ANL_EXT.1	IDS_SDC_EXT.1	Met	
	FPT_STM.1	Met	
IDS_RCT_EXT.1	IDS_SDC_EXT.1	Met	
IDS_RDR_EXT.1	FAU_GEN.1	Met	
IDS_SDC_EXT.1	FPT_STM.1	Met	

9 Acronyms

This section defines the acronyms used in this document.

Table 21 – Acronyms

Acronym	Definition
ACL	Access Control List
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
CC	Common Criteria
CEM	Common Evaluation Methodology
CLI	Command Line Interface
CRC	Cyclic Redundancy Check
CSP	Critical Security Parameter
CTR	Counter
DEK	Data-Encrypting Key
DH	Diffie Hellman
DRBG	Deterministic RBG
DSA	Digital Signature Algorithm
EAL	Evaluation Assurance Level
EEPROM	Electrically Erasable Programmable Read-Only Memory
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standard
GbE	Gigabit Ethernet
GUI	Graphical User Interface
HMAC	Hashed Message Authentication Code
HTTPS	Hypertext Transport Protocol Secure
IDS	Intrusion Detection System
IKE	Internet Key Exchange
IP	Internet Protocol
Ipsec	IP Security
Ipv4	IP version 4
Ipv6	IP version 6
IT	Information Technology
KAT	Known Answer Test

Acronym	Definition
KEK	Key-Encrypting Key
L2TP	Layer Two Tunneling Protocol
LED	Light-Emitting Diode
MB	Megabyte
NAT	Network Address Translation
NIST	National Institute of Standards and Technology
PPTP	Point-to-Point Tunneling Protocol
PSK	Pre-shared Key
PUB	Publication
RAM	Random Access Memory
RBG	Random Bit Generator/Generation
RDSA	RSA Digital Signature Algorithm
RFC	Request for Comment
RSA	Rivest Shamir Adleman
SA	Security Association
SFP	Security Functional Policy
SFR	Security Functional Requirement
SHS	Secure Hash Standard
SNMP	Simple Network Management Protocol
SRAM	Static RAM
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
UTM	Unified Threat Management
VLAN	Virtual Local Area Network
VPN	Virtual Private Network

Prepared by:
Corsec Security, Inc.

13135 Lee Jackson Memorial Highway
Suite 220
Fairfax, VA 22033
United States of America

Phone: +1 (703) 267-6050
Email: info@corsec.com
http://www.corsec.com