# McAfee Network Security Platform (NSP)
# M-Series and NS-Series Sensors
# Version 8.1
# Security Target

| | |
|---|---|
| Release Date: | March 2, 2017 |
| Version: | 1.1 |

Prepared By: Primasec Ltd.

Prepared For: Intel Corporation
2821 Mission College Blvd
Santa Clara, California 95054

# Table of Contents

# List of Tables

# List of Figures

# 1 Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), conformance claims, ST organization and terminology. It also includes an overview of the evaluated product.

## 1.1 Security Target Reference

The Security Target reference shall uniquely identify the Security Target.

| | |
|---|---|
| ST Title: | McAfee Network Security Platform M-Series and NS-Series Sensors Version 8.1 Security Target |
| ST Version Number: | Version 1.1 |
| ST Author(s): | Primasec Ltd. |
| ST Publication Date: | March 2, 2017 |

## 1.2 Target of Evaluation Reference

The Target of Evaluation reference shall identify the Target of Evaluation.

| | |
|---|---|
| TOE Developer | Intel Corporation<br>2821 Mission College Blvd<br>Santa Clara, California 95054 |
| TOE Identification: | McAfee Network Security Platform M-series and NS-Series Sensors |
| TOE Version | 8.1 (M-Series firmware 8.1.15.14 and NS-Series firmware 8.1.17.16) |

The TOE is one or more of the M-Series or NS-Series sensors (listed in Section 1.8.1) running the NSP Sensor software.

## 1.3 Document Organization

**Security Target Introduction (Section 1)** – Provides identification of the TOE and ST, an overview of the TOE, an overview of the content of the ST and relevant terminology. The introduction also provides a description of the TOE security functions as well as the physical and logical boundaries for the TOE, the hardware and software that make up the TOE, and the physical and logical boundaries of the TOE.

**Conformance Claims (Section 2)** – Provides applicable Common Criteria (CC) conformance claims, Protection Profile (PP) conformance claims and Assurance Package conformance claims.

**Security Problem Definition (Section 3)** – Describes the threats, organizational security policies, and assumptions pertaining to the TOE and the TOE environment.

**Security Objectives (Section 4)** – Identifies the security objectives for the TOE and its supporting environment as well as a rationale that objectives are sufficient to counter the threats identified for the TOE.

**Extended Components Definition (Section 5)** – Presents components needed for the ST but not present in Part II or Part III of the Common Criteria Standard.

**Security Requirements (Section 6)** – Presents the Security Functional Requirements (SFRs) met by the TOE and the security functional requirements rationale. In addition this section presents Security Assurance Requirements (SARs) met by the TOE as well as the assurance requirements rationale.

**TOE Summary Specification (Section 7)** – Describes the security functions provided by the TOE that satisfy the security functional requirements, provides the rationale for the security functions.

## 1.4 Common Criteria Product type

The TOE is classified as a **Network Device** (Intrusion Detection System (IDS)).

## 1.5 TOE Overview

The McAfee Network Security Platform (NSP) sensor performs stateful inspection on a packet basis to discover and prevent intrusions, misuse, denial of service (DoS) attacks, and distributed denial of service (DDoS) attacks. McAfee Incorporated offers various types of sensor appliances providing different bandwidth and deployment strategies. These are the models listed in Table 2 (section 1.8.1).

The NSP sensors are managed via a CLI (accessed from a locally connected console or over SSH). Network Security Manager (NSM) software can be used to push policies that have been defined in the Management Component to the appropriate sensor module, but this functionality is outside the scope of this evaluation.

The transfer of signature and profile files between the NSM and NSP sensor is implemented using 128bit AES and SHA-1, with the key being wrapped and sent via the Command Channel.

The NSP sensor’s presence on the network is transparent. The NSP system is protected from the monitored networks as the system is configured to not accept any management requests or input.

An SCP Server is used to transfer updates of the signature files and software of NSP sensors (available from the Update Server that resides at McAfee Incorporated facilities) to the TOE. The platform hosting the NSM may also be used as the SCP Server to host update files.

## 1.6 TOE Description

The NSP IDS product is a combination of network appliances and software built for the detection of intrusions, denial of service (DoS) attacks, distributed denial of service (DDoS) attacks and network misuse. The sensor appliances are stand-alone appliances from McAfee Incorporated.

The NSP Sensor component performs:

*Traffic Capture* captures packets into a data store for review.

*Load balancing and protocol verification* makes security decisions such that it can filter packets of no interest.

*Denial of Service detection and response* detects DoS attacks and provides an alert capability and the capability to drop packets identified that are part of the DoS attack.

*Signature detection and anomaly detection* performs anomaly detection, logs attack information and performs response functions. The response functions include the following: alert generation, packet logging, TCP reset, ICMP host unreachable, forward blocking (Quarantine), alert filtering and dropping of packets. Smart Blocking technology reduces the likelihood of false-positives by identifying attacks and associating a confidence level for each identified attack.

*Sensor management* is used for administration of the TOE by the authorized administrator for basic administration of the sensor (configuration of auditing, firmware update, etc). This also provides the interface between the sensor and the NSM, which can be used to push policies that have been defined in the Management Component to the appropriate sensor module.

The McAfee Incorporated Update Server is a McAfee Incorporated owned and operated file server that makes available updates of the signature files of NSP sensors in customer installations. These new signature files are available through the internet.

The TOE supports SSH and TLS encrypted traffic and enforces all functionality of this traffic on selected sensor models. Through the management interface, the Administrator can import up to 64 certificates to allow the TOE to decrypt and analyze traffic passing through the sensor.

Audit logs can be exported from the NSP Sensor to an audit server over SSH.

The TOE can be configured to perform file-based analysis of traffic in search of malicious payloads. The sensor creates hashes of suspect files being transmitted through the sensor and communicates (through the DNS server) with the Global Threat Intelligence (GTI) File Reputation and IP Reputation database to identify malicious files.

## 1.7 Architecture Description

### 1.7.1 NSP Sensor

The primary function of the NSP sensor (also referred to as the Collector Component) is to analyze traffic on selected network segments and to respond when an attack is detected. The sensor examines the header and data portion of every network packet; scanning for patterns and behavior in the network traffic that indicates malicious activity.

The sensor can operate in three modes:

**Inline**: The product is installed as an appliance within the network that applicable traffic must flow through.

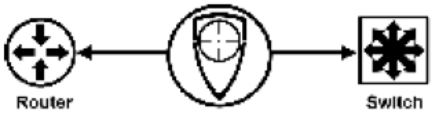


**Figure 1: NSP Sensor in "Inline" mode**

**Tap**: The network traffic flows between the clients and servers and the data is copied by the tap to the sensor which is essentially invisible to the other network entities. Note that the TOE cannot inject response packets back through an external tap, so NSP sensors offer Response ports through which a response packet (such as a TCP reset) can be injected to close a malicious connection.

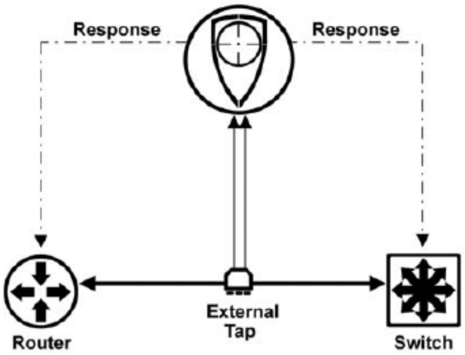


**Figure 2: NSP Sensor in "Tap" mode**

**Span**: The traffic is *spanned* off either the server side or the client side of a router or switch, copying both the incoming and outgoing traffic from any one of the ports. This requires a special network device that has a span port capability. Note that SPAN mode is also a "sniffing" mode, which—unlike inline mode—does not enable the TOE to prevent attacks from reaching their targets. However, while the TOE can issue response packets via the sensor's response ports, some switches allow response packets to be injected by an IPS back through the SPAN port.

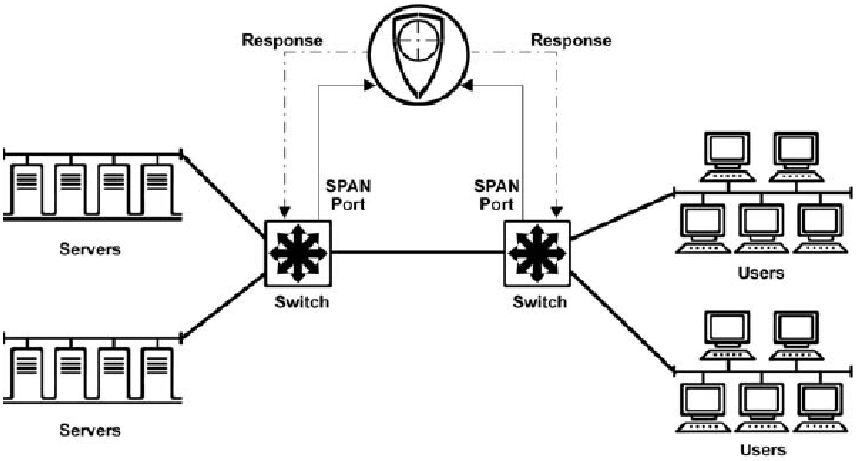


**Figure 3: NSP Sensor in "Span" mode**

A single multi-port NSP sensor can monitor many network segments in any combination of *operating modes*; *monitoring* or *deployment* mode for the sensor; SPAN mode, TAP mode, or INLINE mode.

The NSP's Virtual IDS (VIDS) feature enables you to further segment a port on a sensor into many "virtual sensors". A VIDS can be *dedicated* to a specific network port with monitoring rules appropriate for that segment which might be different than the rules used to monitor other segments. Alternately, if a monitored network segment includes the use of Virtual LANs (VLANs) or Classless Inter-Domain Routing (CIDR); one or more VIDS can be directed at monitoring them with VIDS each configured with distinct monitoring rules. Note that VIDS are not particularly security relevant in and of themselves, but rather serve to organize and distinguish monitoring rules.

### 1.7.2 Operational Environment

#### 1.7.2.1 Network Access

The TOE requires network access to the following support mechanisms in the Operational Environment (outside of the management network):

1. SCP server with an updated image
2. DNS server to support GTI File Reputation and IP Reputation lookups.
3. SMTP server (optional)

No specific versions of the above are required. These operational environment servers are relied on to provide the stated service. No other connections are allowed outside of the management network.

#### 1.7.2.2 Management

Local network access must be allowed to the following resources:

1. McAfee Network Security Manager (NSM) Management Server v8.1 acting as the external audit server
2. SCP server acting as Update Server
3. SNMPv3 client (optional, outside scope of evaluation)

The TOE is certified to run with or without the optional component above.

## 1.8 Physical Boundaries

### 1.8.1 Hardware and Software components

The TOE consists of software developed by McAfee and purpose-built hardware appliances designed by McAfee. In addition, data is to be provided from the environment.

| | Component | TOE/Environment |
|---|---|---|
| NSP Sensor | NSP Sensor Software – M-Series v8.1.15.14<br>NSP Sensor Software –NS-Series v8.1.17.16<br>Installed on McAfee sensor hardware platform | **TOE** |
| Console Workstation | Standard workstation (Windows, Linux, etc.) platform with the ability to initiate console connections. | Environment |

| | | |
|---|---|---|
| Management Workstation | Standard workstation (Windows, Linux, etc.) platform with the ability to initiate SSH connections. | Environment |
| Data | TLS Certificates (private keys) of traffic intended to be decrypted by the TOE | Environment |

**Table 1: TOE and environment hardware & software**

The NSP sensor appliances included in the evaluation are specified in Table 2.

| Model Number | Part Number | Revision |
|---|---|---|
| Sensor M-2750 | 600-1209-03-G | D |
| Sensor M-2850 | 600-1470-03-G | D |
| Sensor M-2950 | 600-1429-01-G | D |
| Sensor M-3050 | 600-1246-06-G | F |
| Sensor M-4050 | 600-1245-06-G | F |
| Sensor M-6050 | 600-1220-07-G | E |
| Sensor M-8000 | 600-1221-07-G (Primary)<br>600-1222-07-G (Secondary) | E<br>E |
| Sensor NS 7100 | 600-1586-01 | G |
| Sensor NS 7200 | 600-1585-01 | G |
| Sensor NS 7300 | 600-1584-01 | G |
| Sensor NS 9100 | 600-1568-04 | A |
| Sensor NS 9200 | 600-1569-04 | A |
| Sensor NS 9300 | 600-1571-04<br>600-1572-04 | A<br>A |

**Table 2: Sensor models**

### 1.8.2 Platforms not included in the TOE

The following NSP sensor appliances are also available, but have not been tested as part of this evaluation, and are excluded from the scope of the TOE.

| Model Number | Part Number | Revision |
|---|---|---|
| Sensor M-1450 | 600-1230-04-G | D |
| Sensor M-1250 | 600-1231-04-G | D |

**Table 3: Sensor models excluded from the scope of the TOE**

### 1.8.3 Guidance Documents

The following table identifies the software components and indicates whether or not each component is in the TOE or the environment.

The following guidance documents are provided for download with the TOE software from the McAfee support website and apply to the CC Evaluated configuration:

#### 1.8.3.1 System Level Guides

1. IPS Administration Guide Revision A, McAfee Network Security Platform 8.1
2. CLI Guide, Revision A, McAfee Network Security Platform 8.1
3. Network Security Platform 8.1 Common Criteria Evaluated Configuration Guide Revision J

#### 1.8.3.2 Applicable Sensor Guides

1. M-6050 Sensor Quick Start Guide Revision B, McAfee® Network Security Platform 8.1
2. M-8000 Sensor Quick Start Guide Revision B, McAfee® Network Security Platform 8.1
3. M-2750 Sensor Quick Start Guide Revision B, McAfee® Network Security Platform
4. M-2850/M-2950 Sensor Quick Start Guide Revision B, McAfee® Network Security Platform
5. M-3050/M-4050 Sensor Quick Start Guide Revision B, McAfee® Network Security Platform
6. NS7x00 Quick Start Guide Revision B, McAfee® Network Security Platform
7. NS9x00 Quick Start Guide Revision B, McAfee® Network Security Platform
8. M-2750 Sensor Product Guide Revision B, McAfee® Network Security Platform
9. M-2850/M-2950 Sensor Product Guide Revision C, McAfee® Network Security Platform
10. M-3050/M-4050 Sensor Product Guide Revision B, McAfee® Network Security Platform
11. M-6050 Sensor Product Guide Revision B, McAfee® Network Security Platform
12. M-8000 Sensor Product Guide Revision C, McAfee® Network Security Platform
13. NS7x00 Sensor Product Guide Revision D, McAfee® Network Security Platform
14. NS9x00 Sensor Product Guide Revision F, McAfee® Network Security Platform

## 1.9 Logical Boundaries

The logical boundaries of the TOE include those security functions implemented exclusively by the TOE.

- Security Audit
- Identification and Authentication
- Security Management
- Protection of the TSF
- User Data Protection

- Cryptographic support

### 1.9.1 Security Audit

The TOE generates audit records related to NSP Sensor operation and administration. These audit records are forwarded by the TOE to the NSM management platform (and stored in a MySQL database) for external retention of the audit logs.

The audit files can be uploaded to the audit server by the administrator. The files are purged after upload. The file size is limited to 10Mb on the M-Series and 128Mb on the NS series. When the file is exhausted, new events are dropped.

Only users authenticated at the TOE CLI (and therefore considered to be authorized administrators) can view audit records.

### 1.9.2 Identification and Authentication

Administrators connecting to the TOE via the Console Port or SSH connection are required to enter a sensor administrator username and password to authenticate the administrative connection prior to access being granted to the CLI.

The NSM is authenticated by the TOE through a shared secret that is configured during the initial installation and setup process of the TOE.

### 1.9.3 Security Management

An administrative CLI can be accessed via the Console port or SSH connection. This interface is used for administration of the TOE, including audit log configuration, upgrade of firmware and signatures, administration of users, configuration of SSH and TLS connections.

Only administrators authenticated to the “Admin” role are considered to be “authorized administrators”.

### 1.9.4 Protection of TSF

The NSP sensors components presence on the network is transparent. The NSP system is protected from the monitored networks as the system is configured to not accept any management requests or input.

The TOE users must authenticate to the TOE before any administrative operations can be performed on the system.

The TOE ensures consistent timestamps are used by receiving time information with its NSM, so that all parts of the NSP system share the same relative time information.

The administrator can query the currently installed versions of software on the NSP sensor using the “show” command, which returns details of the software and hardware version of the sensor. Trusted update of the TOE software can be performed from the command line using the “loadimage scp” command. This loads the image from an SCP server over SSH.

A suite of self tests is performed by the TOE at power on and conditional self-tests are performed continuously.

### 1.9.5 User Data Protection

The NSP sensor ensures the network memory buffers are zeroized prior to allocation to prevent the possibility of residual information from previous connections being used to pad network packets.

### 1.9.6 Cryptographic support

The TOE uses symmetric key cryptography to secure communications between the TOE and the NSM in the Operational Environment for time/date synchronization.

Connections between the TOE and the SCP Server (for firmware update), and between the TOE and audit server (for audit log upload) are secured using SSH and are authenticated using username and password.

Sessions between the Management Workstation and the TOE are secured using SSH, authenticated using username and password, and local console connections between the Console Workstation and the TOE are physically secured.

# 2 Conformance Claims

## 2.1 Common Criteria Conformance Claims

This Security Target is conformant to the Common Criteria Version 3.1 Revision 4, CC Part 3 conformant, CC Part 2 extended.

## 2.2 Conformance to Protection Profiles

This Security Target claims exact conformance to the Protection Profile for Network Devices, Information Assurance Directorate, Version 1.1 [NDPP] and Security Requirements for Network Devices Errata 3, dated 3 November 2014 [NDPPerr3].

"Exact" conformance is defined as the ST containing all of the requirements in section 4 of the [NDPP] & [NDPPerr3], and appropriate requirements from Appendix C of [NDPP]. While iteration is allowed, no additional requirements (from the CC parts 2 or 3) are allowed to be included in the ST. Further, no requirements in section 4 of the [NDPP] are allowed to be omitted. There is no further iteration of requirements from [NDPP] section and there are no additional Part 2 or Part 3 components included in this ST that are not specified in [NDPP] and [NDPPerr3].

## 2.3 Conformance to Security Packages

This Security Target does not claim conformance to any security function requirements or assurance packages, neither as package-conformant or package-augmented.

## 2.4 Conformance claim rationale

The statement of the security problem definition in this ST (Section 3) is the same as that defined in [NDPP].

The statement of the security objectives in this ST (Section 4) is the same as that defined in [NDPP].

The statement of the security requirements in this ST (Section 6) is the same as that defined in [NDPP] and [NDPPerr3].

# 3 Security Problem Definition

The TOE is intended to be used either in environments in which, at most, sensitive but unclassified information is processed, or the sensitivity level of information in both the internal and external networks is equivalent.

This section contains assumptions regarding the security environment and the intended usage of the TOE and threats on the TOE and the IT environment.

## 3.1 Assumptions

The assumptions are taken from [NDPP].

| Short name | Assumption |
|---|---|
| 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. |

**Table 4: Assumptions**

## 3.2 Threats

The TOE or environment addresses the threats identified in this section. The primary assets to be protected are the integrity and availability of the resources and traffic on a network. There is also the concept of the network resources being used in line with organizational policy. The threat agents are authorized persons/processes, unauthorized persons/processes, or external IT entities not authorized to use the TOE itself. The threats identified assume that the threat agent is a person with a basic attack potential who possesses an average expertise, few resources, and low to moderate motivation.

The threats are taken from [NDPP].

| Threat Name | Threat Definition |
|---|---|
| T.ADMIN_ERROR | An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms. |

| | |
|---|---|
| T.TSF_FAILURE | Security mechanisms of the TOE may fail, leading to a compromise of the TSF. |
| 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.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.USER_DATA_REUSE | User data may be inadvertently sent to a destination not intended by the original sender. |

**Table 5: Threats**

## 3.3 Organizational Security Policy

The organizational security policy is taken from [NDPP].

| Policy Name | Policy Definition |
|---|---|
| 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. |

**Table 6: Organizational security policy**

# 4 SECURITY OBJECTIVES

This chapter describes the security objectives for the TOE and the environment. The security objectives are divided between TOE Security Objectives (for example, security objectives addressed directly by the TOE) and Security Objectives for the Environment (for example, security objectives addressed by the IT domain or by non-technical or procedural means).

## 4.1 Security Objectives for the TOE

The table below defines the security objectives that are to be addressed by the TOE.

| TOE Security Objective Name | TOE Security Objective Definition |
|---|---|
| O.PROTECTED_COMMUNICATIONS | The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities. |
| 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. |
| O.SYSTEM_MONITORING | The TOE will provide the capability to generate audit data and send those data to an external IT entity. |
| O.DISPLAY_BANNER | The TOE will display an advisory warning regarding use of the TOE. |
| 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.RESIDUAL_INFORMATION_CLEARING | The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated. |
| O.SESSION_LOCK | The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked. |
| O.TSF_SELF_TEST | The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly. |

**Table 7: Security objectives for the TOE**

## 4.2 Security Objectives for the Environment

The security objectives for the environment listed below are to be satisfied without imposing technical requirements on the TOE (i.e. through procedural, administrative or other technical means):

| TOE Security Objective Name | TOE Security Objective Definition |
|---|---|
| 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. |

**Table 8: Security objectives for the environment**

## 4.3 Mapping of Security Problem Definition to Security Objectives

The following table represents a mapping of the threats, assumptions and organizational security policy to the security objectives defined in this ST.

| | A.NO_GENERAL_PURPOSE | A.PHYSICAL | A.TRUSTED_ADMIN | T.ADMIN_ERROR | T.TSF_FAILURE | T.UNDETECTED_ACTIONS | T.UNAUTHORIZED_ACCES | T.UNAUTHORIZED_UPDAT | T.USER_DATA_REUSE | P.ACCESS_BANNER |
|---|---|---|---|---|---|---|---|---|---|---|
| O.PROTECTED_COMMUNICATIONS | | | | | | | X | | X | |
| O.VERIFIABLE_UPDATES | | | | | | | | X | | |
| O.SYSTEM_MONITORING | | | | | | X | | | | |
| O.DISPLAY_BANNER | | | | | | | | | | X |
| O.TOE_ADMINISTRATION | | | | | | | X | | | |
| O.RESIDUAL_INFORMATION_CLEARING | | | | | | | | | X | |
| O.SESSION_LOCK | | | | | | | X | | | |
| O.TSF_SELF_TEST | | | | | X | | | | X | |
| OE.NO_GENERAL_PURPOSE | X | | | | | | | | | |
| OE.PHYSICAL | | X | | | | | | | | |
| OE.TRUSTED_ADMIN | | | X | X | | | | | | |

**Table 9: Security Problem & IT Security Objectives Mappings**

## 4.4 Rationale for Threat Coverage

This section provides a justification that for each threat, the security objectives counter the threat.

T.ADMIN_ERROR is addressed by OE.TRUSTED_ADMIN. The use of trusted administrators will help to reduce the likelihood of error.

T.TSF_FAILURE is addressed by O.TSF_SELFTEST. Self-testing will reduce the likelihood of undetected failures in TSF mechanisms compromising the security of the TOE.

T.UNDETECTED_ACTIONS is addressed by O.SYSTEM_MONITORING . The TOE will monitor and record selected events (O.SYSTEM_MONITORING).

T.UNAUTHORIZED_ACCESS is addressed by O.PROTECTED_COMMUNICATIONS, O.TOE_ADMINISTRATION and O.SESSION_LOCK. Communication channels are protected against interception (O. PROTECTED_COMMUNICATIONS). Login is controlled (O.

TOE_ADMINISTRATION), and session locking is provided (O.SESSION_LOCK

T.UNAUTHORIZED_UPDATE is addressed by O.VERIFIABLE _UPDATES. The TOE will verify that updates are unaltered (O.VERIFIABLE _UPDATES)

T.USER_DATA_REUSE is addressed by O.PROTECTED_COMMUNICATIONS, O.RESIDUAL_INFORMATION_CLEARING and O.TSF_SELF_TEST. Protection against sending data to an incorrect destination is provided through protection of communication channels (O.PROTECTED_COMMUNICATIONS), clearing of information from objects before reuse (O.RESIDUAL_INFORMATION_CLEARING), and through self testing to ensure correct operation (O.TSF_SELF_TEST).

## 4.5 Rationale for Organizational Security Policy Coverage

P.ACCESS_BANNER requires the display of an access banner. The TOE provides such a banner (O.DISPLAY_BANNER).

## 4.6 Rationale for Assumption Coverage

Each of the assumptions is addressed through provision of a correspondingly named objective for the TOE environment to assure that the assumptions are upheld in the TOE's operational environment.

# 5 Extended Components Definition

## 5.1 Introduction

For this evaluation the Security Functional Requirements in CC Part 2 have been extended to cover part of the TOE functionality that cannot otherwise clearly be expressed.

The extended components in this section are used in [NDPP]. The PP author has not provided definitions of these components, but it is considered appropriate to try to provide these definitions in this ST.

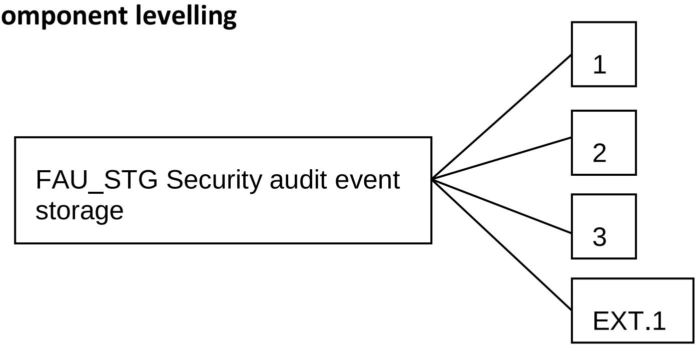
## 5.2 Class FAU: Security Audit

### 5.2.1 Security audit event storage (FAU_STG)

**Family behaviour**

This component is added to the existing family FAU_STG.

**Component levelling**



FAU_STG_EXT.1 requires the ability to transmit or receive audit data to or from a secure external IT entity.

**Management:** FAU_STG_EXT.1

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

a) Management of transmission/receipt of audit data.

**Audit:** FAU_STG_EXT.1

There are no auditable events foreseen.

**FAU_STG_EXT.1 External audit trail storage**

Hierarchical to: No other components

Dependencies: FTP_ITC.1 Inter-TSF trusted channel

FAU_STG_EXT.1.1 The TSF shall be able to [selection: *transmit the generated audit data to an external IT entity, receive and store audit data from an external IT entity*] using a trusted channel implementing the [selection: *SSH, TLS, TLS/HTTPS*] protocol.

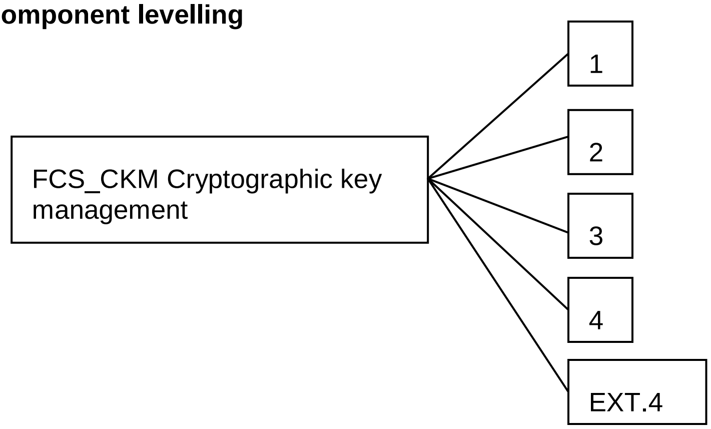
## 5.3 Class FCS: Cryptographic Support

### 5.3.1 Cryptographic key management (FCS_CKM)

**Family behaviour**

This component is added to the existing family FCS_CKM.

**Component levelling**



FCS_CKM_EXT.4 requires the ability to zeroize cryptographic keys and critical security parameters (CSPs).

**Management:** FCS_CKM_EXT.4

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 PP/ST:

a) Minimal: Failure of the activity.

**FCS_CKM_EXT.4 Cryptographic key zeroization**

Hierarchical to: No other components

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

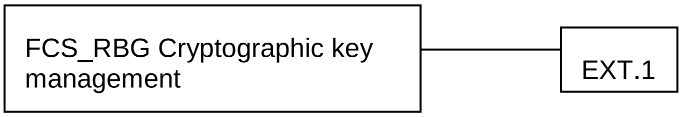
FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

### 5.3.2 Cryptographic operation: random bit generation (FCS_RBG_EXT)

**Family behaviour**

This family is added to the class FCS. This family deals with generation of random bit streams in support of cryptographic operations

**Component levelling**



FCS_RBG_EXT.1 requires generation of random bits in accordance with a selected standard.

**Management:** FCS_RBG_EXT.1

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 PP/ST:

a) Minimal: Failure of the activity.

**FCS_RBG_EXT.1Cryptographic operation: random bit generation**

Hierarchical to: No other components

Dependencies: No dependencies

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with [selection, choose one of: *NIST Special Publication 800-90 using* [selection: *Hash_DRBG (any), HMAC_DRBG (any), CTR_DRBG (AES), Dual_EC_DRBG (any)]; FIPS Pub 140-2 Annex C: X9.31 Appendix 2.4 using AES*] seeded by an entropy source that accumulated entropy from [selection: *a software-based noise source, a TSF-hardware-based noise source*].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded with a minimum of [selection: *128 bits, 256 bits*] of entropy at least equal to the greatest security strength of the keys and hashes that it will generate.

### 5.3.3 SSH (FCS_SSH_EXT)

**Family behaviour**

This family is added to the class FCS, and places specific requirements on the implementation of SSH.

Component levelling

FCS_SSH_EXT.1 places specific requirements on the implementation of SSH.

Management: FCS_SSH_EXT.1

No management activities are foreseen.

Audit: FCS_SSH_EXT.1

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

a) Minimal: Failure to establish an SSH session,

b) Basic: Establishment and termination of an SSH session.

**FCS_SSH_EXT.1 SSH**

Hierarchical to: No other components

Dependencies: No dependencies

FCS_SSH_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, and 4254, and [selection: 5656, 6668, no other RFCs].

FCS_SSH_EXT.1.2 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.1.3 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.1.4 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.1.5 The TSF shall ensure that the SSH transport implementation uses [selection: SSH-RSA , ecdsa-sh2-nistp256] and [selection: PGP-SIGN-RSA, PGP-SIGN-DSS, ecdsa-sha2-nistp384, no other public key algorithms] as its public key algorithm(s).

FCS_SSH_EXT.1.6 The TSF shall ensure that data integrity algorithms used in SSH transport connection is [selection: hmac-sha1, hmac-sha1-96, hmac-sha2-256, hmac-sha2-512].

FCS_SSH_EXT.1.7 The TSF shall ensure that diffie-hellman-group14-sha1 and [selection: ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521, no other methods] are the only allowed key exchange methods used for the SSH protocol.

### 5.3.4 TLS (FCS_TLS)

**Family behaviour**

This family is added to the class FCS, and places specific requirements on the implementation of TLS.

**Component levelling**

FCS_TLS_EXT.1 places specific requirements on the implementation of TLS.

**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 PP/ST:

a) Minimal: Failure to establish a TLS session,

b) Basic: Establishment and termination of a TLS session.

**FCS_TLS_EXT.1 TLS**

Hierarchical to: No other components

Dependencies: No dependencies

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:

Mandatory Ciphersuites

TLS_RSA_WITH_AES_128_CBC_SHA

Optional Ciphersuites:

[selection:

*None*

TLS_RSA_WITH_AES_256_CBC_SHA

TLS_DHE_RSA_WITH_AES_128_CBC_SHA

TLS_DHE_RSA_WITH_AES_256_CBC_SHA

*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*].

## 5.4 Class FIA: Identification and Authentication

### 5.4.1 Password management (FIA_PMG)

**Family behaviour**

This family is added to the class FIA, and deals with the specification of rules for password composition.

**Component levelling**

FIA_PMG Password management
EXT.1

FIA_PMG_EXT.1 requires that passwords should conform to rules that are configurable by the system administrator.

**Management:** FIA_PMG_EXT.1

There are no management activities foreseen.

**Audit:** FIA_PMG_EXT.1

There are no auditable events foreseen.

**FIA_PMG_EXT.1 Password management**

Hierarchical to: No other components

Dependencies: FIA_UAU_EXT.2 Password-based authentication mechanism

FIA_PMG_EXT.1.1 The TSF shall provide the following password management capabilities for administrative passwords:

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters [selection: *“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”*,[assignment: *other characters*]];

b) Minimum password length shall be settable by the security administrator, and support passwords of 15 characters or greater.

### 5.4.2 User identification and authentication (FIA_UIA)

**Family behaviour**

This family is added to the class FIA, and combines aspects of the existing CC families FIA_UID and FIA_UAU.

Component levelling

FIA_UIA_EXT User identification and authentication
EXT.1

FIA_UIA_EXT.1 allows for specification of a limited set of actions to be permitted before a user is identified and authenticated.

**Management:** FIA_UIA_EXT.1

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

a) Management of the user identities;

b) Management of the authentication data by an administrator;

c) Management of the authentication data by the associated user;

b) If an authorised administrator can change the actions allowed before identification and authentication, the managing of the action lists.

**Audit:** FIA_UIA_EXT.1

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

a) Minimal: Unsuccessful use of the authentication mechanism;

b) Basic: All use of the authentication mechanism;

c) Detailed: All TSF mediated actions performed before identification and authentication of the user.

**FIA_UIA_EXT.1 User identification and authentication**

Hierarchical to: No other components

Dependencies: No dependencies

FIA_UIA_EXT.1.1 The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

- Display the warning banner in accordance with FTA_TAB.1;
- [selection: *no other actions*, [assignment: *list of services, actions performed by the TSF in response to non-TOE requests*]].

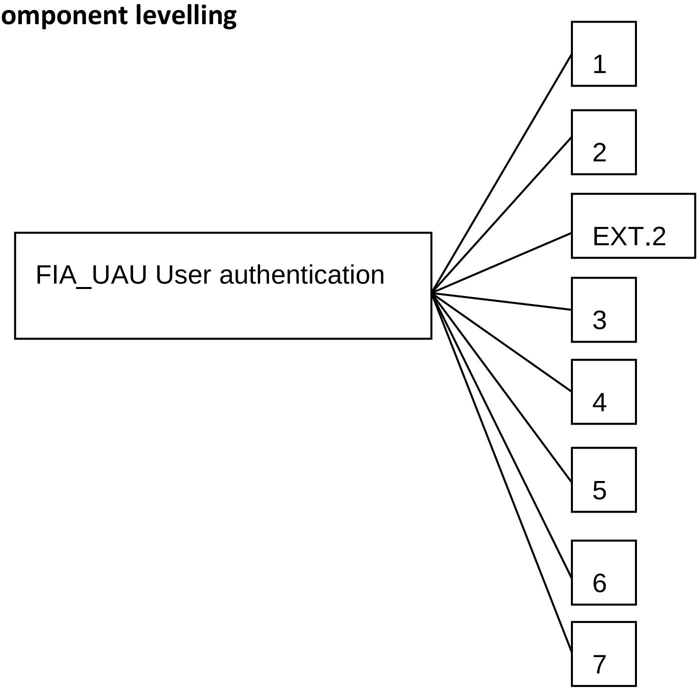
FIA_UIA_EXT.1.2 The TSF shall require each user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that administrative user.

### 5.4.3 User authentication (FIA_UAU)

**Family behaviour**

This component is added to the existing CC family FIA_UAU, and covers use of a password for authentication.

**Component levelling**



FIA_UAU_EXT.2 allows for specification of password based and other authentication mechanisms.

**Management:** FIA_UAU_EXT.2

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

a) Resetting of the expired passwords.

**Audit:** FIA_UAU_EXT.2

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

a) Minimal: Unsuccessful use of the authentication mechanism;

b) Basic: All use of the authentication mechanism.

**FIA_UAU_EXT.2 Password-based authentication mechanism**

Hierarchical to: No other components

Dependencies: FIA_PMG_EXT.1 Password management

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [selection: [assignment: *other authentication mechanism(s)*], *none*] to perform administrative user authentication.

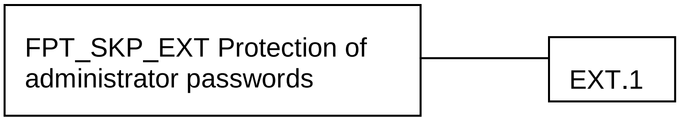
## 5.5 Class FPT: Protection of the TSF

### 5.5.1 Protection of TSF data (FPT_SKP)

**Family behaviour**

This family is added to the class FPT, and addresses the requirement to prevent reading of sensitive TSF data.

**Component levelling**



FPT_SKP_EXT.1 requires that sensitive cryptographic keys cannot be read.

**Management:** FPT_SKP_EXT.1

There are no management activities foreseen.

**Audit:** FPT_SKP_EXT.1

There are no auditable events foreseen.

**FPT_SKP_EXT.1 Protection of TSF data (for reading of all symmetric keys)**

Hierarchical to: No other components

Dependencies: No dependencies

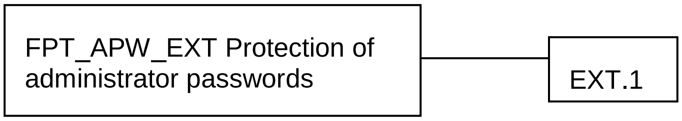
FPT_SKP_EXT.1.1 The TSF shall prevent reading of all pre-shared keys, symmetric keys and private keys.

### 5.5.2 Protection of administrator passwords (FPT_APW)

**Family behaviour**

This family is added to the class FPT, and addresses the requirement to prevent reading of plaintext passwords.

**Component levelling**



FPT_APW_EXT.1 requires that passwords are not stored in clear, and that no interface is provided to read them.

**Management:** FPT_APW_EXT.1

There are no management activities foreseen.

**Audit:** FPT_APW_EXT.1

There are no auditable events foreseen.

**FPT_APW_EXT.1 Protection of administrator passwords**

Hierarchical to: No other components

Dependencies: No dependencies

FPT_APW_EXT.1.1 The TSF shall store passwords in non-plaintext form.

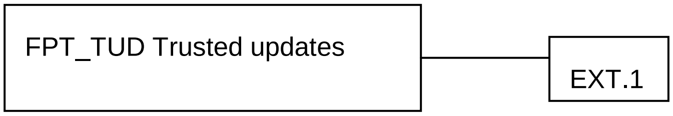
FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext passwords.

### 5.5.3 Trusted update (FPT_TUD)

**Family behaviour**

This family is added to the class FPT, and addresses the requirement to query the current version of the TOE, and to initiate and verify updates.

**Component levelling**



FPT_TUD_EXT.1 requires the ability to query the current TOE version, and to initiate and verify updates.

**Management:** FPT_TUD_EXT.1

There are no management activities foreseen.

**Audit:** FPT_TUD_EXT.1

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

a) Minimal: Initiation of any update to the TOE software/firmware.

**FPT_TUD_EXT.1 Trusted update**

Hierarchical to: No other components

Dependencies: No dependencies

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.

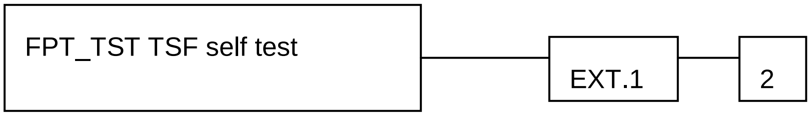
FPT_TUD_EXT.1.3 The TSF shall provide a means to verify firmware/software updates to the TOE using a [selection: *digital signature mechanism, published hash*] prior to installing those updates.

### 5.5.4 TSF self test (FPT_TST)

**Family behaviour**

This component is added to the existing CC family FPT_TST.

**Component levelling**



FPT_TUD_EXT.1 requires the ability to query the current TOE version, and to initiate and verify updates.

**Management:** FPT_TST_EXT.1

There are no management activities foreseen.

**Audit:** FPT_TST_EXT.1

There are no auditable events foreseen.

**FPT_TST_EXT.1 TSF testing**

Hierarchical to: No other components

Dependencies: No dependencies

FPT_TST_EXT.1.1 The TSF shall run a suite of self tests [selection: *during initial start-up (on power on), periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which a self test should occur]*] to demonstrate the correct operation of [selection: *[assignment: parts of the TSF], the TSF*].

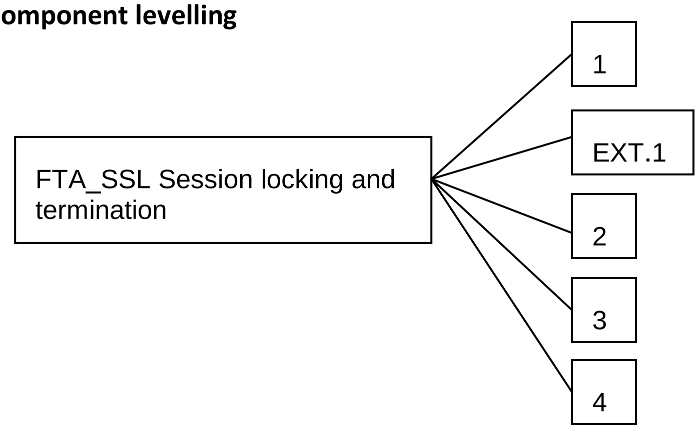
## 5.6 Class FTA: TOE Access

### 5.6.1 Session locking and termination (FTA_SSL)

**Family behaviour**

This component is added to the existing CC family FTA_SSL.

**Component levelling**



FTA_SSL_EXT.1 requires the ability to either lock or terminate a local interactive session.

**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 or termination occurs for an individual user;

b) Specification of the default time of user inactivity after which lock-out or termination occurs;

c) Management of the events that should occur prior to unlocking the session.

**Audit:** FTA_SSL_EXT.1

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

a) Basic: Any attempts at unlocking a locked interactive session.

**FTA_SSL_EXT.1 TSF-initiated session locking**

Hierarchical to: No other components

Dependencies: FIA_UIA_EXT.1 User identification and authentication

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, [selection: *lock the session – disable any 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.

# 6 Security Requirements

The security requirements that are levied on the TOE are specified in this section of the ST. These security requirements are defined in Sections 6.2

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4*, with additional extended functional components.

## 6.1 Conventions

The CC defines four operations on security functional requirements. The conventions below define the conventions used in this ST to identify these operations, as also applied in [NDPP].

Assignment: indicated with *italicized text*

Selection: indicated with underlined text

Assignment within selection: indicated with *italicized and underlined text*

Refinement: additions indicated with **bold** text and ~~strike-through~~ if necessary

Iteration: indicated with typical CC requirement naming followed by a lower case letter for each iteration (e.g. FMT_MSA.1a)

The explicitly stated requirements claimed in this ST (denoted by the EXT suffix) are taken from [NDPP] and are defined in Section 5 above.

The operations completed in [NDPP] and [NDPPerr2] are shown in this ST in the same manner as shown in [NDPP] and [NDPPerr2], even where they are not fully compliant with the conventions specified in those documents.

## 6.2 TOE Security Functional Requirements

The SFRs defined in this section are taken from Part 2 of the CC and from the definition of extended components provided in Section 5 above.

### 6.2.1 Security Audit (FAU)

#### 6.2.1.1 FAU_GEN.1 Audit Data Generation

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

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

b) All auditable events for the not specified level of audit; and

c) *All administrative actions*;

d) *[Specifically defined auditable events listed in Table 10*].

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FAU_GEN.1 | None. | None |
| FAU_GEN.2 | None. | None |
| FAU_STG_EXT.1 | None. | None |
| FCS_CKM.1 | None | None |
| FCS_CKM_EXT.4 | None | None |
| FCS_COP.1(1) | None | None |
| FCS_COP.1(2) | None | None |
| FCS_COP.1(3) | None | None |
| FCS_COP.1(4) | None | None |
| FCS_RBG_EXT.1 | None | None |
| FCS_SSH_EXT.1 | Failure to establish a SSH session | Reason for failure |
| | Establishment/Termination of a SSH session | Non-TOE endpoint of connection (IP address) for both successes and failures |
| FCS_TLS_EXT.1 | Failure to establish a TLS session | Reason for failure |
| | Establishment/Termination of a TLS session | Non-TOE endpoint of connection (IP address) for both successes and failures |
| FDP_RIP.2 | None. | None |
| FIA_PMG_EXT.1 | None. | None |
| FIA_UIA_EXT.1 | All use of the identification and authentication mechanism. | Provided user identity, origin of the attempt (e.g., IP address). |
| FIA_UAU_EXT.2 | All use of the authentication mechanism. | Origin of the attempt (e.g., IP address). |
| FIA_UAU.7 | None | None |
| FMT_MTD.1 | None | None |
| FMT_SMF.1 | None | None |
| FMT_SMR.2 | None | None |
| FPT_SKP_EXT.1 | None | None |
| FPT_APW_EXT.1 | None | None |
| FPT_STM.1 | Changes to the time. | The old and new values for the time.<br>Origin of the attempt (e.g., IP address). |
| FPT_TUD_EXT.1 | Initiation of update. | No additional information. |
| FPT_TST_EXT.1 | None | None |
| FTA_SSL_EXT.1 | Any attempts at unlocking of an interactive session[1]. | No additional information. |
| FTA_SSL.3 | The termination of a remote | No additional information. |

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| | session by the session locking mechanism. | |
| FTA_SSL.4 | The termination of an interactive session | No additional information. |
| FTA_TAB.1 | None. | None |
| FTP_ITC.1 | Initiation of the trusted channel.<br>Termination of the trusted channel.<br>Failure of the trusted channel functions. | Identification of the initiator and target of failed trusted channels establishment attempt. |
| FTP_TRP.1 | Initiation of the trusted channel.<br>Termination of the trusted channel.<br>Failures of the trusted path functions. | Identification of the claimed user identity. |

**Table 10: SFRs and their auditable actions**

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, and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*information specified in column three of Table 10*].

#### 6.2.1.2 FAU_GEN.2 User Identity Association

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.

#### 6.2.1.3 FAU_STG_EXT.1 External Audit Trail Storage

FAU_STG_EXT.1.1 The TSF shall be able to [transmit the generated audit data to an external IT entity] using a trusted channel implementing the [SSH] protocol.

### 6.2.2 Cryptographic support (FCS)

Application Note: [NDPP] does not specify that correct cryptographic operation must be validated through compliance with FIPS 140. However, the Canadian Common Criteria Scheme requires that this is done, and so compliance with FIPS 140 is considered implicit in the following cryptographic requirements. Certificate numbers are provided in section 7.4.

#### 6.2.2.1 FCS_CKM.1 Cryptographic Key Generation (for asymmetric keys)

FCS_CKM.1.1 **Refinement:** The TSF shall generate **asymmetric** cryptographic keys **used for key establishment** in accordance with [

- NIST Special Publication 800-56B, “Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography” for RSA-based key establishment schemes] and specified cryptographic key sizes *equivalent to, or greater than, a symmetric key strength of 112 bits.*

### 6.2.2.2 FCS_CKM_EXT.4 Cryptographic Key Zeroization

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

### 6.2.2.3 FCS_COP.1(1) Cryptographic Operation (for data encryption/decryption)

FCS_COP.1.1(1) **Refinement:** The TSF shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm [*AES operating in* [*CBC mode*]] and cryptographic key sizes 128-bits, 256-bits, and [no other key sizes] that meet~~s~~ the following:

- **FIPS PUB 197, "Advanced Encryption Standard (AES)"**
- [NIST SP 800-38A].

### 6.2.2.4 FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)

FCS_COP.1.1(2) **Refinement:** The TSF shall perform **cryptographic signature services** in accordance with a

1) [RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater]

that meets the following:

Case: RSA Digital Signature Algorithm

- **FIPS PUB 186-4**, "Digital Signature Standard"].

Application Note: While [NDPP] provides only FIPS PUB 186-2 or FIPS PUB 186-3 as selection operations, the FIPS guidance is for the adoption of FIPS PUB 186-4 and this is considered to be an appropriate refinement of the FCS_COP.1.1(2) requirement.

### 6.2.2.5 FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)

FCS_COP.1.1(3) **Refinement:** The TSF shall perform [*cryptographic hashing services*] in accordance with a specified cryptographic algorithm [SHA-256] **and message digest sizes** [256] **bits** that meet the following: *FIPS Pub 180-3, "Secure Hash Standard"*.

### 6.2.2.6 FCS_COP.1(4) Cryptographic Operation (for keyed-hash message authentication)

FCS_COP.1.1(4) **Refinement:** The TSF shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm HMAC-[SHA1]**, key size** [*128 bits*]**, and message digest sizes** [160] **bits** that meet the following: *FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code, and FIPS Pub 180-3, "Secure Hash Standard"*.

### 6.2.2.7 FCS_RBG_EXT.1 Extended: Cryptographic Operation (Random Bit Generation)

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with [NIST Special Publication 800-90 using [CTR_DRBG]] seeded by an entropy source that accumulated entropy from [a 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 security strength of the keys and hashes that it will generate.

### 6.2.2.8 FCS_SSH_EXT.1 SSH

FCS_SSH_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [no other RFCs].

FCS_SSH_EXT.1.2 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.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*256k*] bytes in an SSH transport connection are dropped.

FCS_SSH_EXT.1.4 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.1.5 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.1.6 The TSF shall ensure that data integrity algorithms used in SSH transport connection is [hmac-sha1, hmac-sha1-96].

FCS_SSH_EXT.1.7 The TSF shall ensure that diffie-hellman-group14-sha1 and [no other methods] are the only allowed key exchange methods used for the SSH protocol.

### 6.2.2.9 FCS_TLS_EXT.1 TLS

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:

**Mandatory Ciphersuites**

TLS_RSA_WITH_AES_128_CBC_SHA

**Optional Ciphersuites**:

[None].

## 6.2.3 User Data Protection (FDP)

### 6.2.3.1 FDP_RIP.2 Full Residual Information Protection

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

## 6.2.4 Identification and Authentication (FIA)

### 6.2.4.1 FIA_PMG_EXT.1 Password Management

FIA_PMG_EXT.1.1 The TSF shall provide the following password management capabilities for administrative passwords:

a) *Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters* [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [*None*]];

*b) Minimum password length shall settable by the Security Administrator, and support* *passwords of 15 characters or greater*~~;~~**.**

#### 6.2.4.2 FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1.1 The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

- Display the warning banner in accordance with FTA_TAB.1;
- [no other actions].

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

#### 6.2.4.3 FIA_UAU_EXT.2 Extended: Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [none] to perform administrative user authentication.

#### 6.2.4.4 FIA_UAU.7 Protected Authentication Feedback

FIA_UAU.7.1 The TSF shall provide only *obscured feedback* to the administrative user while the authentication is in progress at the local console.

### 6.2.5 Security Management (FMT)

#### 6.2.5.1 FMT_MTD.1 Management of TSF Data (for general TSF data)

FMT_MTD.1.1 The TSF shall restrict the ability to *manage* the *TSF data* to the *Security Administrators*.

#### 6.2.5.2 FMT_SMF.1 Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to update the TOE, and to verify the updates using [digital signature] capability prior to installing those updates;*
- [No other capabilities].

#### 6.2.5.3 FMT_SMR.2 Restrictions on security roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- **Authorized Administrator**.

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the conditions

- **Authorized Administrator role shall be able to administer the TOE locally;**
- **Authorized Administrator role shall be able to administer the TOE remotely;**

are satisfied.

Application Note: Only those administrators authenticated to accounts assigned to the NSP Sensor role “Admin” are considered to be *authorized administrators*.

## 6.2.6 Protection of the TSF (FPT)

### 6.2.6.1 FPT_SKP_EXT.1 Extended: Protection of TSF Data (for reading of all symmetric keys)

FPT_SKP_EXT.1.1 The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

### 6.2.6.2 FPT_APW_EXT.1.1 Extended: Protection of administrator passwords

FPT_APW_EXT.1.1 The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext passwords.

### 6.2.6.3 FPT_STM.1 Reliable Time Stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

### 6.2.6.4 FPT_TUD_EXT.1 Extended: Trusted Update

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] prior to installing those updates.

### 6.2.6.5 FPT_TST_EXT.1 Extended: TSF Testing

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.

## 6.2.7 TOE Access (FTA)

### 6.2.7.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions,

- [terminate the session]

after a Security Administrator-specified time period of inactivity.

### 6.2.7.2 FTA_SSL.3 TSF-initiated Termination

FTA_SSL.3.1 **Refinement:** The TSF shall terminate **a remote** interactive session after a [*Security Administrator-configurable time interval of session inactivity*].

### 6.2.7.3 FTA_SSL.4 User-initiated termination

FTA_SSL.4.1 The TSF shall allow Administrator-initiated termination of the Administrator’s own interactive session.

#### 6.2.7.4 FTA_TAB.1 Default TOE Access Banners

FTA_TAB.1.1 **Refinement:** Before establishing **an administrative user** session, the TSF shall display **a Security Administrator-specified** advisory **notice and consent** warning message regarding use of the TOE.

### 6.2.8 Trusted Path/Channels (FTP)

#### 6.2.8.1 FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1 **Refinement:** The TSF shall **use** [TLS, SSH] **to** provide a **trusted** communication channel between itself and **authorized IT entities supporting the following capabilities: audit server,** [*SCP Server acting as update server*, *NSM*] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data **from disclosure and detection of modification of the channel data**.

FTP_ITC.1.2 The TSF shall permit *the TSF*, ***or the authorized IT entities*** to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [*all trusted communications with an IT peer*].

Application Note: Audit logs are uploaded to the audit server (NSM) over SSH. TOE updates are transferred from the SCP Server over SCP connections, which are protected using SSH. Date/time synchronization is performed when a TLS connection is established with the NSM.

#### 6.2.8.2 FTP_TRP.1 Trusted Path

FTP_TRP.1.1 **Refinement:** The TSF shall **use** [SSH] **to** provide a **trusted** communication path between itself and **remote administrators** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *disclosure and detection of modification of the communicated data.*

FTP_TRP.1.2 The TSF shall permit **remote administrators** to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for *initial administrator authentication and all remote administrative actions*.

## 6.3 TOE Security Assurance Requirements

The assurance requirements listed below are those in [NDPP], providing compliance with the protection profile. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

| Assurance Class | Assurance Components | |
|---|---|---|
| ADV: Development | ADV_FSP.1 | Basic functional specification |
| AGD: Guidance documents | AGD_OPE.1 | Operational user guidance |

| Assurance Class | Assurance Components | |
|---|---|---|
| | AGD_PRE.1 | Preparative procedures |
| ALC: Life-cycle support | ALC_CMC.1 | Labeling of the TOE |
| | ALC_CMS.1 | TOE CM coverage |
| ASE: Security target evaluation | ASE_CCL.1 | Conformance claims |
| | ASE_ECD.1 | Extended components definition |
| | ASE_INT.1 | ST introduction |
| | ASE_OBJ.2 | Security objectives |
| | ASE_REQ.2 | Derived security requirements |
| | ASE_SPD.1 | Security problem definition |
| | ASE_TSS.1 | TOE summary specification |
| ATE: Tests | ATE_IND.1 | Independent testing - conformance |
| AVA: Vulnerability assessment | AVA_VAN.1 | Vulnerability survey |

**Table 11: Assurance Requirements**

## 6.4 Rationale for TOE Security Requirements

### 6.4.1 TOE Security Functional Requirements

| Security Objective | Mapping Rationale |
|---|---|
| O.PROTECTED_ COMMUNICATIONS | Communications protection is provided through use of encrypted services for data transfer (FTP_ITC.1), and for administrator sessions (FTP_TRP.1). These services are supported by functions to manage encryption/decryption (FCS_COP.1(1)), key generation and management (FCS_CKM.1, FCS_CKM_EXT.4, FCS_RBG_EXT.1, FPT_SKP_EXT.1), digital signature FCS_COP.1(2), and hashing (FCS_COP.1(3), FCS_COP.1(4)). Specific services are provided for TLS and SSH (FCS_TLS_EXT.1, FCS_SSH_EXT.1). Encryption functions can be configured by an administrator (FMT_SMF.1). |
| O.VERIFIABLE_UPDATES | The TOE provides functionality to generate hash values (FMT_SMF.1, FCS_COP.1(3)) that can be used only by an administrator to check the validity of updates (FPT_TUD_EXT.1). |
| O.SYSTEM_MONITORING | The TOE generates audit records (FAU_GEN.1) that are attributable to users (FAU_GEN.2). Audit records may be exported for storage (FAU_STG_EXT.1). |

| Security Objective | Mapping Rationale |
|---|---|
| O.DISPLAY_BANNER | The TOE generates a warning banner following login (FTA_TAB.1). |
| O.TOE_ADMINISTRATION | The TOE controls login (FIA_UIA_EXT.1) using passwords (FIA_PMG_EXT.1) that are not stored in clear (FPT_APW_EXT.1). Entered passwords are not displayed on screen (FIA_UAU.7). Protection is provided through session expiry (FTA_SSL_EXT.1, FTA_SSL.3), and protection of communication paths against modification or disclosure (FTP_TRP.1). A number of security management roles are defined (FMT_SMR.2), and the ability to manage TSF data is restricted (FMT_MTD.1). |
| O.RESIDUAL_INFORMATION_CLEARING | The TOE provides clearing of resources on allocation (FDP_RIP.2). |
| O.SESSION_LOCK | The TOE provides the capability to lock a terminate session following a period of inactivity (FTA_SSL_EXT.1), and also to terminate remote sessions after a period of inactivity (FTA_SSL.3, FTA_SSL.4). |
| O.TSF_SELF_TEST | The TOE runs a suite of self-tests following power on (FPT_TST_EXT.1). |

**Table 12: Security objective mapping rationale**

### 6.4.2 TOE Security Assurance Requirements

The TOE SARs are consistent with the threat environment, and are taken from [NDPP].

## 6.5 Rationale for IT security functional requirement dependencies

This section includes a table of all the security functional requirements and their dependencies and a rationale for any dependencies.

| Functional Component | Dependency | Included/Rationale |
|---|---|---|
| FAU_GEN.1 | FPT_STM.1 | Yes |
| FAU_GEN.2 | FAU_GEN.1, FIA_UID.1 | FAU_GEN.1 is included. Dependency on FIA_UID.1 met by FIA_UIA_EXT.1, which includes that functionality. |
| FAU_STG_EXT.1 | FTP_ITC.1 | Yes |
| FCS_CKM.1 | FCS_CKM.2 or FCS_COP.1, FCS_CKM.4 | Yes, through FCS_COP.1 and FCS_CKM_EXT.4 |
| FCS_CKM_EXT.4 | FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 | Yes, using FCS_CKM.1 |

| Functional Component | Dependency | Included/Rationale |
|---|---|---|
| FCS_COP.1(1) | FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1, FCS_CKM.4 | Yes, using FCS_CKM.1 and FCS_CKM_EXT.4 (which provides equivalent functionality to FCS_CKM.4) |
| FCS_COP.1(2) | FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1, FCS_CKM.4 | Yes, using FCS_CKM.1 and FCS_CKM_EXT.4 (which provides equivalent functionality to FCS_CKM.4) |
| FCS_COP.1(3) | FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1, FCS_CKM.4 | Yes, using FCS_CKM.1 and FCS_CKM_EXT.4 (although dependencies are not relevant as this component relates to hashing only) |
| FCS_COP.1(4) | FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1, FCS_CKM.4 | Met using FCS_CKM.1 and FCS_CKM_EXT.4 (which provides equivalent functionality to FCS_CKM.4) |
| FCS_RBG_EXT.1 | None | Yes |
| FCS_TLS_EXT.1 | None | Yes |
| FDP_RIP.2 | None | Yes |
| FIA_PMG_EXT.1 | FIA_UAU_EXT.2 | Yes |
| FIA_UIA_EXT.1 | None | Yes |
| FIA_UAU_EXT.2 | FIA_PMG_EXT.1 | Yes |
| FIA_UAU.7 | FIA_UAU.1 | Dependency is met using FIA_UIA_EXT.1 |
| FMT_MTD.1 | FMT_SMR.2, FMT_SMF.1 | Yes |
| FMT_SMF.1 | None | Yes |
| FMT_SMR.2 | FIA_UID.1 | Dependency is met using FIA_UIA_EXT.1 |
| FPT_APW_EXT.1 | FIA_UAU_EXT.2 | Yes |
| FPT_SKP_EXT.1(2) | FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 | Yes, using FPT_CKM.1 |
| FPT_STM.1 | None | Yes |
| FPT_TUD_EXT.1 | None | Yes |
| FPT_TST_EXT.1 | None | Yes |
| FTA_SSL_EXT.1 | FIA_UIA_EXT.1 | Yes |
| FTA_SSL.3 | None | Yes |
| FTA_SSL.4 | None | Yes |
| FTA_TAB.1 | None | Yes |

| Functional Component | Dependency | Included/Rationale |
|---|---|---|
| FTP_ITC.1 | None | Yes |
| FTP_TRP.1 | None | Yes |

**Table 13: SFR dependencies**

# 7 TOE Summary Specification

The TOE's security functionality is characterized through the following Security Functions:

- Security Audit
- Identification and Authentication
- Security Management
- Protection of the TSF
- User Data Protection

## 7.1 Security Audit

FAU_GEN.1, FAU_GEN.2

The TOE generates audit records for operation and administration of the sensor. Audit events are recorded in auditlog. In addition, sshlog must be enabled to ensure ssh events are recorded in auditlog.

The auditlog file size is limited to 10Mb on M-Series and 128Mb on NS series. The file is purged from the disk when the audit log is uploaded and a new auditlog file is started with a start marker. The file can also be removed from the disk by the administrator using the CLI command line. When the file is exhausted, new events are dropped and a sysevent is sent to the NSM informing no further audit messages will be recorded until the log is purged.

Events logged in audit records include the items listed in Table 10, plus start-up and shut-down of the audit functions.

The following information about an audited event is stored in the audit log whenever that audited event is recorded:

a) Date and time of the event,

b) Type (i.e., category and action) of event,

c) Subject (i.e., user and domain) identity,

d) Result (success or failure) of the event, and

e) Description (where applicable access mode, target object, etc.).

FAU_STG_EXT.1

The audit functions resulting in events being recorded in the auditlog can be enabled/disabled by authorized administrators using the following commands:

- "set auditlog"
- "set sshlog"

Use of the "set auditlog" and "set sshlog" commands is audited. The tracelog auditing is always enabled and cannot be disabled.

No facility is provided on the TOE to view or modify the audit log file, as the CLI is provided by a zebra shell, which provides no filesystem access and only a limited set of commands and does not support a “root” user. So the log file cannot be directly access by any authorized administrator.

There is no filesystem access to any administrative users (as the CLI is provided by a zebra shell with a limited set of commands).

## 7.2 Identification and Authentication

FIA_UAU_EXT.1, FTA_TAB.1

Identification and authentication is required for both local and remote administrator access. Remote access to the TOE is via an SSH session (from the Management Workstation) and local access to the TOE is via the appliance console port (from the Console Workstation), both of which provide access to the TOE CLI.

Prior to logon via the NSP sensor console port and SSH connection a consent banner is displayed for to the administrator warning that proceeding with authentication is consenting to the terms of use of the TOE. Having acknowledged the access banner, the user is then prompted to enter their username and password at the command line prompt. The NSP sensor uses the kernel authentication mechanism, storing the username and SHA256 hash of the password in the file /etc/passwd and /etc/shadow. The NSP sensor performs a SHA256 hash of the password entered by the user and performs a lookup in of the username and hash value of the password in the /etc/passwd and /etc/shadow files. If the credentials correspond to an entry in the files the user is successfully authenticated and is authorized to access the NSP Sensor CLI, and a command prompt is presented to the user.

FIA_UAU_EXT.2, FIA_UAU.7, FIA_PMG_EXT.1

Authentication of an administrator is configured to be through use of a username/password. The profile will specify a minimum of 15 characters, that incorporate a combination of lowercase letters, uppercase letters, numbers and special characters (“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”). During entry of the password, each character entered is masked with a “*” when progress is reflected on the screen. If an authentication attempt fails, (either the username is not recognized or the password is incorrect) the same “Login failed” error message is presented.

If successful, the user’s session is initiated under the assigned role. If unsuccessful, the authentication attempt fails and connection is immediately terminated.

## 7.3 Security Management

FMT_MTD.1, FMT_SMF.1, FMT_SMR.2, FPT_TUD_EXT.1

The TOE CLI provides a security management interface used to configure and manage the sensor. Only authorized administrators (those users assigned to the sensor role “Admin”) are able to access this interface to manage the configuration of the TOE (as enforced by the Identification and Authentication function, detailed in Section 7.2).

Following successful authentication authorized administrators are able to perform management actions such as query the current version of the TOE software of the NSP sensor and can initiate an update of the TOE software from the SCP Server (which gets its updates from the Update Server in the TOE environment). The sensor images are signed with a McAfee key. Once the image has been downloaded, the sensor checks the signature of the image (against the McAfee public key stored in a plaintext file in the sensor internal media; Compact flash for M-Series and SSF on NS-Series) before the image is applied.

FTA_SSL_EXT.1, FTA_SSL.3, FTA_SSL.4

Following an administrator configured period of inactivity (of both local and remote interactive session) the session will be terminated, requiring re-authentication by the administrator before the access to TOE functionality can be gained.

The administrator is able to issue a "logoff" option to terminate their session once they have completed all administrative tasks.

## 7.4 Protection of the TSF

FCS_CKM.1, FCS_CKM_EXT.4, FCS_RBG_EXT.1, FCS_TLS_EXT.1, FCS_SSH_EXT.1, FCS_COP.1(1), (2), (3) & (4), FTP_ITC.1, FTP_TRP.1

The TOE incorporates in its own cryptographic modules the OpenSSL v2.0.5 library (modified to comply with FIPS 186-4 RSA) and the XySSL v1.1.2.2 library for provision of all cryptographic services and operations. The NS-series and M-series firmware cryptographic modules have undergone assessment against the FIPS140-2 criteria, including their key generation and AES encryption functionality. The NS7x00-series, NS9x00-series and M-series sensors are FIPS 140-2 validated, as shown in Table 14 below.

| Platform | FIPS 140 Cert # |
|---|---|
| M-Series | 2555 |
| M-8000 P | 2558 |
| M-8000 S | 2572 |
| NS 7100/7200/7300 | 2842 |
| NS 9100/9200 | 2591 |
| NS 9300 P | 2596 |
| NS 9300 S | 2593 |

**Table 14: CMVP module certificate numbers**

The TOE uses the services provided by the (modified) OpenSSL Module integrated in the TOE's N-series and MS-series FIPS modules to provide all cryptographic services and operations, with the exception of the POST Software Integrity test, which is supported by the XySSL library.

OpenSSL is used to generate asymmetric cryptographic keys using a domain parameter generator and a random bit generator that meet Block Cipher (CTR) DRBG, compliant with SP800-90A, with an equivalent key strength of at least 112 bits (2048 bits for RSA keys), using AES. The FIPS approved OpenSSL PRNG is used by the NSP sensors, which is seeded through /dev/random (with fips_mode_set() enabled). /dev/random serves as a true random number generator that captures environmental noise collected from device drivers and other module sources. Input from these sources is added to an entropy pool, which is mixed using a CRC-like function, and are extracted using a SHA hash of the pool to avoid exposing the internal state of the pool. The pool provides 512 bytes of random data.

The TOE uses symmetric key cryptography to secure communications between the TOE and external IT. All sessions are protected using TLSv1 (conforming to RFC 2246) or SSH (conforming to the mandatory portions of RFCs 4251, 4252, 4253, and 4254[2]).The connections between the TOE and the SCP Server (acting as the update server), between the TOE and the audit server, and between the TOE and the remote administrator are secured using SSH. The Sensor communicates with the NSM management platform (which is used to host the SCP Server and audit server in the evaluated configuration) and with the administrator through its dedicated 10M/100M management Ethernet port. The Sensor can also communicate with the administrator through its dedicated console port from the Console Workstation. This console port is a serial port and as such is only used by administrators with physical access to the Sensor, on a directly connected serial cable between the Sensor and the Console Workstation.

SSH authentication is achieved using password based authentication for Management Workstation, audit server and SCP Server session establishment. This method of SSH authentication is performed in accordance with [RFC4252] 'Password Authentication Method: "password"'. Diffie Hellman group 14 is the only key exchange method supported by the TOE for RSA public/private key exchange in the SSH session establishment.

In accordance with [RFC4252] the encryption algorithm used for SSH transport is either AES-CBC-128 or AES-CBC-256 as negotiated during the session establishment. Similarly, the data integrity algorithm applied is either hmac-sha1 or hmac-sha1-96 as negotiated during the session establishment. It is hardcoded in the TOE that these are the algorithms that will be accepted – the TOE will accept no others – and the key length (128 or 256) used for the session is dependent on which the SCP Server or SSH client on the Management Workstation will accept. There is no administrator configuration required to specify supported algorithms.

If the TOE receives an SSH packet that exceeds 256K, the packet is silently dropped, with no response to the originator of the packet.

Sessions between the TOE and the NSM Manager (used to synchronize date/time for timestamps) are secured over a TLS tunnel (using AES, 128 bit encryption, ciphersuite TLS_RSA_WITH_AES_128_CBC_SHA). Message integrity checking is provided by a SHA1 hash.

The RSA private keys used for SSH public key based authentication and for the TLS_RSA_WITH_AES_128_CBC_SHA ciphersuite are stored in a plaintext file on internal media (Compact flash for M-Series and SSD on NS-Series).

CAVP certificate numbers for the algorithms employed are provided in Table 15 below.

| | M-Series | | NS-Series | |
|---|---|---|---|---|
| | OpenSSL-Sensor | XySSL-Sensor | OpenSSL-Sensor | XySSL-Sensor |
| AES | 3155 | n/a | 3156 | n/a |
| RSA 186-4 | 1598 | 1824 | 1600 | 1825 |
| SHA-256 | 2610 | 2922 | 2612 | 2923 |
| HMAC-SHA1 | 1988 | n/a | 1989 | n/a |
| Block Cipher (CTR) DRBG | 648 | n/a | 649 | n/a |

**Table 15: CAVP algorithm certificate numbers**

The following table specifies the zeroization method of cryptographic keys generated and managed on the NSP Sensor.

| NSP Sensor Key | Algorithm | Description of usage and storage | Zeroisation method |
|---|---|---|---|
| RSA Public/private keys (persistent) | Block Cipher (CTR) DRBG/RSA | Plaintext self-signed certificates used for TLS, stored in EEPROM and RAM | Zeroized from EEPROM on resetconfig or NetBoot, Zeroized from RAM on reboot<br>Using OpenSSL scrubbing method |
| SSH Host Private Key | The first time SSH is configured, the key is generated. Used to identify the host. | Plaintext session keys stored in RAM used for SSH host identification | Zeroized from Flash on resetconfig or NetBoot, Zeroized from RAM on reboot.<br>Using OpenSSL scrubbing method |
| SSH Host Public Key | RSA 2048 bit key used to authenticate sensor to remote client during SSH | Plaintext public key stored in RAM | Zeroized in RAM on reboot using OpenSSL scrubbing method |
| SSH Remote Client Public Key | RSA 2048 bit key used to authenticate remote client to sensor during SSH | Plaintext public key stored in RAM | Zeroized in RAM on reboot using OpenSSL scrubbing method |
| SSH Session Key | Session keys used with SSH, AES 128, | Plaintext session keys stored in RAM | Zeroized in RAM on reboot using OpenSSL scrubbing |

| NSP Sensor Key | Algorithm | Description of usage and storage | Zeroisation method |
|---|---|---|---|
| | 256, HMAC-SHA-1 key (160), DH Private Key 2048 | used for SSH session agreement | method |
| Diffie-Hellman key pairs | Block Cipher (CTR) DRBG/DH | Plaintext session keys stored in RAM used for TLS session agreement | Memory scrubbed using OpenSSL method upon termination of session |
| User password | User generated | Plain text value held in RAM as entered by user. | RAM scrubbed using OpenSSL method following completion of authentication request |
| Block Cipher (CTR) DRBG | | Plaintext seed key and state of RNG held in RAM | Memory scrubbed by OpenSSL once seed passed to RNG.<br>RNG scrubbed using OpenSSL method during normal shutdown. |
| McAfee Image Verification Key | RSA 2048 bit key used to authenticate firmware images | Loaded into sensor and stored in EEPROM | |

**Table 16: NSP Sensor Key zeroisation**

Domain parameters used in RSA-based key establishment schemes meet NIST Special Publication 800-56B and key generation is performed in accordance with FIPS186-4. These keys are used in support of the digital signature operations described under FCS_COP.1(2). All of the “shall” and “shall not” statements within SP 800-56B are adhered to. Table 17 provides justification of deviation from the “should” and “should not” statements in the NIST Special Publication.

| Section # | Compliance | Deviation | Rationale for deviation |
|---|---|---|---|
| 5.6 | should | No | n/a |
| 6.1 | should not | No | n/a |
| 6.1 | should (first occurrence) | No | n/a |
| 6.1 | should (second occurrence) | No | n/a |
| 6.1 | should (third occurrence) | No | n/a |

| Section # | Compliance | Deviation | Rationale for deviation |
|---|---|---|---|
| 6.1 | should (fourth occurrence) | No | n/a |
| 6.2.3 | should | No | n/a |
| 6.5.1 | should | No | n/a |
| 6.5.2 | should | No | n/a |
| 6.5.2.1 | should | No | n/a |

**Table 157: 800-56B Compliance**

FPT_SKP_EXT.1, FPT_APW_EXT.1

The OpenSSL container on the NSP Sensor is used to protect the cryptographic keys stored on the sensor. The Sensor 'admin' and 'operator' passwords are stored on the Sensor in the Linux shadow file, protected using a SHA-256 hash. The other Sensor administrative user ('support', 'private' and nobrik1n') passwords are stored in shell.conf, protected using a SHA-256 hash. There is no filesystem access to any administrative users.

FPT_STM.1

The sensor platform maintains a system clock used to provide date/time details for use by the TOE. The NSM management platform (in the TOE environment) periodically passes a timestamp reference to the sensors to ensure clocks within an NSP system are consistent. This occurs: On power up when establishing the TLS crypto channels to NSM, upon every TLS re-establishment due to link/network issues to NSM and when a TLS reconnection is initiated by the administrator. This timestamp is sent by the NSM over a TLS connection.

Each Sensor uses this timestamp to synchronize its own independent timing mechanism synchronizing at regular intervals per the timestamps sent from the NSM management platform. The system clock is also used by the sensor to timestamp all audit events recorded in the audit log, as identified in Section 7.1.

FPT_TST_EXT.1

At power-on a suite of known answer tests are performed to confirm the correct operation of the cryptographic algorithms, together with firmware integrity testes and critical functions tests.

The integrity of all firmware modules is tested using a stored hash. This includes testing of every component in the image. Grub verifies kernel and kernel verifies the rest of the components with an RSA 2048 bit signature. Critical function testing is performed for all available cores to ensure they come up, and if not, the sensor goes "bad state". All hardware components are verified. The firmware integrity tests are supported by the XySSL library RSA digital signature services and SHA hashing services.

Conditional self-tests are performed continuously during operation to confirm continued operation of the DRNG & NDRNG and sign/verify RSA pairwise consistency.

Entropy health testing is performed at start-up and continuously during operation. At shutdown, a set of random data is stored which is used on the next start up. Then a continuous RBG test is performed each time random data is requested. If the test fails due to insufficient entropy in the pool then the function does not provide the entropy, backs off and retries again giving time for additional entropy to be collected.

These tests are sufficient to demonstrate the TSF is operating correctly, as they confirm the integrity of all firmware modules prior to their execution thereby confirming the modules have not been modified or replaced in any unauthorized manner and they ensure the DRNG & NDRNG continue to operate successfully providing sufficient entropy in response to any requests.

## 7.5 User Data Protection

FDP_RIP.2

On the NSP Sensor, network memory traffic is is a collection of packets that are internally managed as a memory buffer (mbuf). mbuf is intiialized during a mempool creation with null/zero content. Once each node (pkt) is processed/dispatched, the metadata associated with the pkt is zerioized in memory. This voids meaningful reconstruction of content.

# 8 Document Terminology

Please refer to CC v3.1 Part 1 Section 4 for definitions of commonly used CC terms.

### 8.1.1 ST Specific Terminology

Authorized Administrator(s)

A general term used in this ST to refer to administrative users holding the “Admin” sensor role.

Attack

A set of actions performed by an attacker that poses a threat to the security state of a protected entity in terms of confidentiality, integrity, authenticity, availability, authorization and access policies.

Denial of Service

In a Denial of Service (DoS) attack, the attacker attempts to crash a service (or the machine), overload network links, overload the CPU, or fill up the disk. The attacker does not always try to gain information, but to simply act as a vandal to prevent the user from making use of the machine.

Distributed DoS (DDoS)

These attacks usually consist of standard DoS attacks orchestrated by attackers covertly controlling many, sometimes hundreds of different machines.

Intrusion

Unauthorized access to, and/or activity in, an information system, usually for the purpose of tampering with or disrupting normal services. See also Attack.

Intrusion Detection

The process of identifying that an intrusion was attempted is in process or has occurred.

MySQL Database

A Relational database that allows for the definition of data structures, storage/retrieval operations and integrity constraints. The data and relations between them are kept in organized tables, which are collections of records and each record in a table contains the same fields.

Roles

A class of user privileges that determines the authorized activities of the various users in the system.

Sensor

The sensor is a network device containing the intrusion detection engine. It analyzes network traffic and searches for signs of unauthorized activity.

Signature

Activities or alterations to an information system indicating an attack or attempted attack, detectable by examination of audit trail logs.

Span Mode

One of the monitoring modes available for an NSP sensor. In the SPAN mode, the sensor functions by mirroring the packet information on a switch or hub and sending the information to a sensor for inspection, while continuing the transmission of traffic with negligible latency. SPAN mode is typically halfduplex, and works through a connection of a sensor to a port on a hub or the SPAN port of a switch.

SPAN Port
On a switch, SPAN mirrors the traffic at one switched segment onto a predefined port, known as a SPAN port.

TLS
Transport Layer Security (TLS) is a cryptographic protocol used for communications over networks. TLS is used to encrypt segments of a network.

Tap
A tap is hardware device that passes traffic unidirectionally from a network segment to the IDS. Traffic is mirrored as it passes through the tap. This mirror image is sent to the IDS for inspection. This prevents traffic passing from being directed at the IDS.

Tap Mode
One of the monitoring modes available for an NSP sensor. In this mode, the NSP functions by mirroring the packet information and sending the information to a sensor for inspection, while continuing the transmission of traffic with negligible latency. Tap mode works through installation of an external wire tap, a port on a hub, or the SPAN port of a switch. This is also known as passive monitoring mode.

VLAN
Virtual Local Area Network. A logical grouping of two or more nodes which are not necessarily on the same physical network segment, but which share the same network number. This is often associated with switched Ethernet networks.

Vulnerability
A weakness that allows an attacker to reduce a system's information assurance by exploiting a system susceptibility or flaw.

Exclusive OR
XOR is a logical operator that results in true if one of the operands (not both) is true.

### 8.1.2 Acronyms

| | |
|---|---|
| CC | Common Criteria |
| CSP | Critical Security Parameters |
| DAC | Discretionary Access Control |
| DoS | Denial of Service |
| EAL | Evaluation Assurance Level |
| FIPS | Federal Information Processing Standards Publication 140-2 |
| GTI | Global Threat Intelligence |
| IDS | Intrusion Detection System |
| IPS | Intrusion Prevention System |
| I/O | Input/Output |

| | |
|---|---|
| MIB | Management Information Base |
| NIST | National Institute of Standards and Technology |
| NSM | Network Security Manager |
| NSP | Network Security Platform |
| PP | Protection Profile |
| SF | Security Functions |
| SFR | Security Functional Requirements |
| ST | Security Target |
| TOE | Target of Evaluation |
| TSF | TOE Security Functions |