Security Target Junos OS 19.2R1 for SRX1500, SRX4100, SRX4200 and SRX4600 Series

---

Juniper Networks

Version 3.7

October 15, 2019

Prepared for:
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
www.juniper.net

Abstract

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), Junos OS 19.2R1 for SRX1500, SRX4100, SRX4200 and SRX4600. This Security Target (ST) is conformant to the requirements of Collaborative Protection Profile for Network Devices (or NDcPP) v2.1, and those for Firewall [FWcPP], Intrusion Prevention Systems [IPS_EP], and VPN Gateway [VPN_EP].

References

[CC1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017
[CC2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017
[CC3]	Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017
[CC_Add]	CC and CEM Addenda, Exact Conformance, Selection-Based SFRs, Optional SFRs, CCDB-2017-05-xxx, Version 0.5, May 2017
[FWcPP]	collaborative Protection Profile for Stateful Traffic Filter Firewalls, Version 2.0+Errata 20180314, dated 14 March 2018
[FWcPP_SD]	Supporting Document, Evaluation Activities for Stateful Traffic Filter Firewall cPP, October 2017, version 2.0
[IPS_EP]	collaborative Protection Profile for Network Devices/collaborative Protection Profile for Stateful Traffic Filter Firewalls Extended Package (EP) for Intrusion Prevention Systems (IPS), version 2.11, dated 15 June 2017
[NDcPP]	Collaborative Protection Profile for Network Devices, version 2.1 dated 24 September 2018
[SD]	Supporting Document, Evaluation Activities for Network Device cPP, September 2018, version 2.1
[VPN_EP]	Network Device Collaborative Protection Profile (NDcPP)/Stateful Traffic Filter Firewall Collaborative Protection Profile (FWcPP) Extended Package VPN Gateway, version 2.1, dated 08 March 2017

Product Guide References

[ECG]	Junos OS Common Criteria Guide for SRX1500, SRX4100, SRX4200 and SRX4600 Devices, Release 19.2R1, 09-Oct-2019
[IDPGuide]	Junos OS Intrusion Detection and Prevention Feature Guide for Security Devices, 13-Oct-19
[VPNGuide]	Junos OS IPSec VPN Feature Guide for Security Devices, 01-Oct-19
[CLIGuide]	Junos OS CLI User Guide, 25-Sep-19
[InsGuide]	Junos OS Installation and Upgrade Guide, 30-Oct-17
[FWGuide]	Junos OS Routing Policies, Firewall Filters, and Traffic Policers Feature Guide, 11-Oct-19
[HW1500]	SRX1500 Services Gateway Hardware Guide, 31-Mar-19
[HW4100]	SRX4100 Services Gateway Hardware Guide, 28-Aug-19
[HW4200]	SRX4200 Services Gateway Hardware Guide, 28-Aug-19
[HW4600]	SRX4600 Services Gateway Hardware Guide, 16-Apr-19
[AAGuide]	Junos OS User Access and Authentication Feature Guide, 5-Aug-19

Table of Contents

1 Introduction

1. This section identifies the Security Target (ST), Target of Evaluation (TOE), Security Target organization, document conventions, and terminology. It also includes an overview of the evaluated products.

1.1 ST reference

ST Title	Security Target Junos OS 19.2R1 for SRX1500, SRX4100, SRX4200 and SRX4600 Series
ST Revision	3.7
ST Draft Date	October15, 2019
Author	Juniper Networks, Inc.
cPP/EP Conformance	[NDcPP], [FWcPP], [VPN_EP], [IPS_EP]

1.2 TOE Reference

TOE Title	Junos OS 19.2R1 for SRX1500, SRX4100, SRX4200 and SRX4600

1.3 About this document

2. This Security Target follows the following format:

Section	Title	Description
1	Introduction	Provides an overview of the TOE and defines the hardware and firmware that make up the TOE as well as the physical and logical boundaries of the TOE
2	Conformance Claims	Lists evaluation conformance to Common Criteria versions, Protection Profiles, or Packages where applicable
3	Security Problem Definition	Specifies the threats, assumptions and organizational security policies that affect the TOE
4	Security Objectives	Defines the security objectives for the TOE/operational environment and provides a rationale to demonstrate that the security objectives satisfy the threats
5	Security Functional Requirements	Contains the functional requirements for this TOE
6	Security Assurance Requirements	Contains the assurance requirements for this TOE
7	TOE Summary Specification	Identifies the IT security functions provided by the TOE and also identifies the assurance measures targeted to meet the assurance requirements

Table 1 Document Organization

1.4 Document Conventions

3. This document follows the same conventions as those applied in [NDcPP] in the completion of operations on Security Functional Requirements, namely:

- Unaltered SFRs are stated in the form used in [CC2] or their extended component definition (ECD);
- Refinement made in the : the refinement text is indicated with **bold text** and ~~strikethroughs~~;
- Selection completed in the ST: the selection values are indicated with <u>underlined text</u>

  e.g. "[*selection: disclosure, modification, loss of use*]" in [CC2] or an ECD might become "<u>disclosure</u>" (completion;
- Assignment completed in the ST: indicated with *italicized text*;

- Assignment completed within a selection in the ST: the completed assignment text is indicated with *__italicized and underlined text__*

  e.g. "[selection: *change_default, query, modify, delete, [assignment: other operations*]]" in [CC2] or an ECD might become "__change_default, *select_tag*__" (completion of both selection and assignment);

- Iteration: indicated by adding a string starting with "/" (e.g. "FCS_COP.1/Hash").

1.5 TOE Overview

4. The Target of Evaluation (TOE) is Juniper Networks, Inc. Junos OS 19.2R1 for SRX1500, SRX4100, SRX4200 and SRX4600, which primarily supports the definition of and enforces information flow policies among network nodes. The Services Gateway appliances provide for stateful inspection of every packet that traverses the network and provide central management to manage the network security policy. All information flow from one network node to another passes through an instance of the TOE. Information flow is controlled on the basis of network node addresses, protocol, type of access requested, and services requested. In support of the information flow security functions, the TOE ensures that security-relevant activity is audited, that their own functions are protected from potential attacks, and provides the security tools to manage all of the security functions. The TOE provides multi-site virtual private network (VPN) gateway functionality. The TOE also implements Intrusion Prevention System functionality, capable of monitoring information flows to detect potential attacks based on pre-defined attack signature and anomaly characteristics in the traffic.

5. All the SRX Services Gateway appliance models run the same Juniper Networks Junos operating system (Junos OS), Junos OS 19.2R1.

6. The Junos OS running on the TOE is implemented as a kernel-based based virtual machine (KVM) running on a hypervisor that's powered by Wind River Linux.

1.6 TOE Description

1.6.1 Overview

7. Each Juniper Networks SRX Services Gateway appliance is a security system that supports a variety of high-speed interfaces (up to 40 Gbps for firewall and 20 Gbps for IPS) for medium/large networks and network applications. Juniper Networks routers share common Junos firmware, features, and technology for compatibility across platforms.

8. The appliances are physically self-contained, housing the software, firmware and hardware necessary to perform all router functions. The hardware has two components: the Services Gateway appliance itself and various PIC/PIMs, which allow the appliances to communicate with the different types of networks that may be required within the environment where the Services Gateway appliances are used.

9. Each instance of the TOE consists of the following major architectural components:

   - The Routing Engine (RE) runs the Junos firmware and provides Layer 3 routing services and network management for all operations necessary for the configuration and operation of the TOE and controls the flow of information through the TOE, including Network Address Translation (NAT) and all operations necessary for the encryption/decryption of packets for secure communication via the IPSec protocol.

   - The Packet Forwarding Engine (PFE) provides all operations necessary for transit packet forwarding.

10. The Routing Engine and Packet Forwarding Engine perform their primary tasks independently, while constantly communicating through a high-speed internal link. This arrangement provides streamlined forwarding and routing control and the capability to run Internet-scale networks at high speeds.

11. The Services Gateway appliances support numerous routing standards for flexibility and scalability as well as IETF IPSec protocols. These functions can all be managed through the Junos firmware, either from a connected terminal console or via a network connection. Network management can be secured using IPsec and SSH protocols. All management, whether from a user connecting to a terminal or from the network, requires successful authentication. In the evaluated deployment Network management is secured using the SSH protocol, which can be tunnelled over IPsec.

12. The TOE supports intrusion detection and prevention functionality, which allows it to detect and react to potential attacks in real time. The detection component of the IPS can be based on attack signatures which specify the characteristics of the potentially malicious traffic based on a variety of packet headers payload data attributes. Anomaly detection based on deviation of the monitored traffic from expected values is also supported.

13. Services Gateway appliances accomplish routing through a process called a Virtual Router (VR). A security device divides its routing component into two or more VRs with each VR maintaining its own list of known networks in the form of a routing table, routing logic, and associated security zones.

14. In the evaluated deployment the TOE is managed and configured via Command Line Interface, either via a directly connected console or using SSH connections (over IPsec).

1.6.2 Physical boundary

15. The physical boundary of the TOE is the entire chassis of the Services Gateway appliance, and so includes both the hardware and firmware of the network device. The TOE is the Junos OS 19.2R1 firmware running on the appliance chassis listed below. This includes the firmware implementing the Routing Engine and the ASICs implementing the Packet Forwarding Engine). Hence the TOE is contained within the physical boundary of the specified appliance chassis. Separate install packages are provided for SRX1500 and SRX4100/SRX4200/SRX4600, namely:

- SRX1500:
  - junos-srxentedge-x86-64-19.2R1.8.tgz
- SRX4100/SRX4200:
  - junos-srxmr-x86-64-19.2R1.8.tgz
- SRX4600:
  - junos-srxhe-x86-64-19.2R1.8.tgz

16. The physical boundary of the above SRX appliance instances of the TOE includes the KVM Hypervisor, which provides the virtualization layer in which Junos OS VM executes.

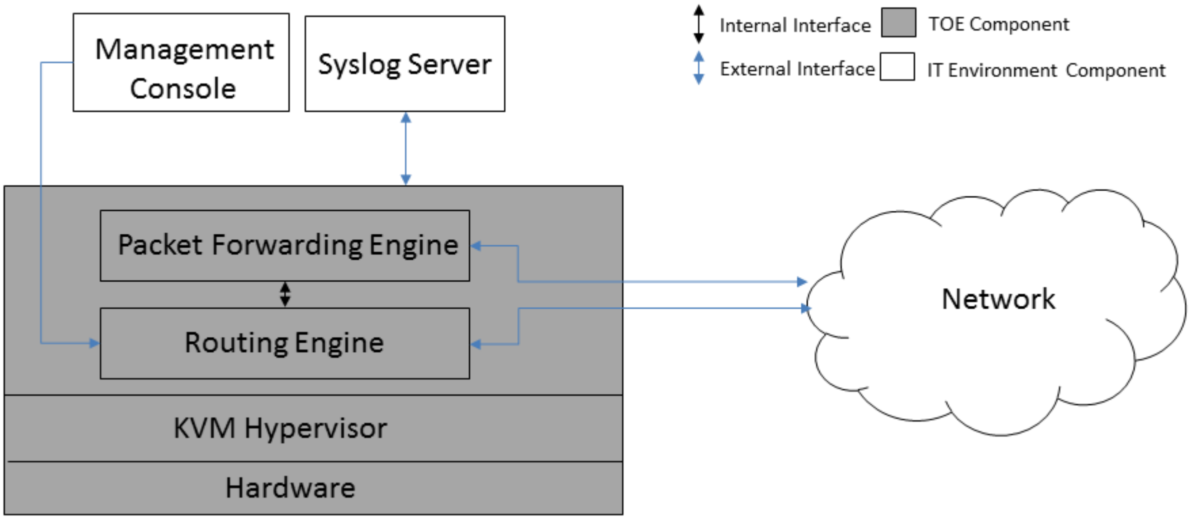

Figure 1 TOE Boundary

17. The TOE interfaces are comprised of the following:

i. Network interfaces which pass traffic

ii. Management interface through which handle administrative actions.

Model	Network Ports	Firmware
SRX1500	• 12x1GbE Ethernet LAN ports • 4x1GbE SFP ports • 4x10GbE SFP+ ports • 1x1GbE Out-of-Band (OOB) management port • 1x1GbE (SFP) Dedicated high availability (HA) ports • 2xPIM slots	Junos OS 19.2R1
SRX4100	• 8x1GbE/10GbE SFP+ ports • 1x1GbE Out-of-Band (OOB) management port • 2x1GbE/10GbE (SFP/SFP+) Dedicated high availability (HA) ports	
SRX4200	• 8x1GbE/10GbE SFP+ ports • 1x1GbE Out-of-Band (OOB) management port • 2x1GbE/10GbE (SFP/SFP+) Dedicated high availability (HA) ports	
SRX4600	• 8x110GbE SFP+ ports • 1x1GbE Out-of-Band (OOB) management port • 2x1GbE/10GbE (SFP/SFP+) Dedicated high availability (HA) ports	

Table 2 TOE Physical Boundary details

18. The guidance documents included as part of the TOE are: [ECG], [IDPGuide], [VPNGuide], [CLIGuide], [InsGuide], and [FWGuide].

1.6.3 Logical Boundary

19. The logical boundary of the TOE includes the following security functionality:

Security Functionality	Description
Protected Communications	The TOE provides an SSH server to support protected communications for administrators to establish secure sessions and to connect to external syslog servers. The TOE also supports IPsec connections to provide multi-site virtual private network (VPN) gateway functionality and also as a tunnel for remote administrate SSH connections. The TOE requires that applications exchanging information with it are successfully authenticated prior to any exchange (i.e. applications connecting over SSH and IPsec). Telnet, File Transfer Protocol (FTP), and Secure Socket Layer (SSL) are out of scope. The TOE includes a baseline cryptographic module that provides the underlying cryptographic services, including key management and protection of stored keys, algorithms, random bit generation and crypto-administration. The cryptographic module provides confidentiality and integrity services for authentication and for protecting communications with adjacent systems.
Administrator Authentication	Administrative users must provide unique identification and authentication data before any administrative access to the system is granted. Authentication data entered and stored on the TOE is protected. The TOE can be configured to terminate interactive user sessions and to present an access banner with warning messages prior to authentication.
Correct Operation	The TOE provides for both cryptographic and non-cryptographic self-tests, and is capable of automated recovery from failure states.
Trusted Update	The administrator can initiate update of the TOE firmware. The integrity of any firmware updates is verified prior to installation of the updated firmware.
Audit	Junos auditable events are stored in the syslog files on the appliance, and can be sent to an external log server (via Netconf over SSH). Auditable events include start-up and shutdown of the audit functions, authentication events, service requests, IPS events, as well as the events listed in Table 4 and Table 5. Audit records include the date and time, event category, event type, username, and the outcome of the event (success or failure). Local syslog storage limits are configurable and are monitored. In the event of storage limits being reached the oldest logs will be overwritten.
Management	The TOE provides a Security Administrator role that is responsible for: • the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the evaluated product • the regular review of all audit data; • initiation of trusted update function; • administration of VPN, IPS and Firewall functionality; • all administrative tasks (e.g., creating the security policy). The devices are managed through a Command Line Interface (CLI). The CLI is accessible through local (serial) console connection or remote administrative (SSH) session.

Security Functionality	Description
Packet Filtering/Stateful Traffic Filtering	The TOE provides stateful network traffic filtering based on examination of network packets and the application of information flow rules.
Intrusion Prevention	The TOE can be configured to analyze IP-based network traffic forwarded to the TOE's interfaces, and detect violations of administratively-defined IPS policies. The TOE is capable of initiating a proactive response to terminate/interrupt an active potential threat, and to initiate a response in real time that would cause interruption of the suspicious traffic flow.
User Data Protection/Information Flow Control	The TOE is designed to forward network packets (i.e., information flows) from source network entities to destination network entities based on available routing information. This information is either provided directly by TOE users or indirectly from other network entities (outside the TOE) configured by the TOE users. The TOE has the capability to regulate the information flow across its interfaces; traffic filters can be set in accordance with the presumed identity of the source, the identity of the destination, the transport layer protocol, the source service identifier, and the destination service identifier (TCP or UDP port number).

1.6.4 Non-TOE hardware/software/firmware

20. SFPs are required by the SRX1500, SRX4100, SRX4200 and SRX4600 models of the TOE to operate, communicate with the connected network. These are detailed for each TOE appliance in Section 10.

21. The TOE relies on the provision of the following items in the network environment:

- Syslog server supporting SSHv2 connections to send audit logs;
- SSHv2 client for remote administration;
- Serial connection client for local administration;
- IPsec peer.

1.6.5 Summary of out scope items

- Use of telnet, since it violates the Trusted Path requirement set (see Section 5.7.2)
- Use of FTP, since it violates the Trusted Path requirement set (see Section 5.7.2)
- Use of SNMP, since it violates the Trusted Path requirement set (see Section 5.7.2)
- Use of SSL, including management via J-Web, JUNOScript and JUNOScope, since it violates the Trusted Path requirement set (see Section 5.7.2)
- Use of CLI account super-user and Linux root account.

2 Conformance Claim

2.1 CC Conformance Claim

22. This Security Target conforms to the requirements of Common Criteria v3.1, Revision 5 and is Part 2 extended and Part 3 conformant.

2.2 PP Conformance claim

23. This Security Target claims Exact Conformance to [NDcPP] (v2.1) and [FWcPP] (v2.0E). Exact conformance is defined in [NDcPP] section 2 and in [CC_Add]. In the rules specified in [CC_Add] it is possible to claim exact conformance to more than one Protection Profile, provided they both require exact conformance. This is the case for both [NDcPP] and [FWcPP].

24. In lieu of any available PP-modules facilitating the expression of functionality provided by this class of network device, this Security Target conforms to the NIAP Extended Packages [IPS_EP] (v2.11) and [VPN_EP] (v2.1). This approach is supported by the fact that [VPN_EP] and [IPS_EP] also require exact conformance, as is required by [CC_Add] when adding requirements from a Package or PP-Module (in this case [VPN_EP] and [IPS_EP]) to Base-PPs (in this case [NDcPP] and [FWcPP]).

25. The Security Problem definition in this Security Target is consistent with the security problem definitions detailed in the collaborative Protection Profiles and Extended Packages to which this ST claims conformance, namely:

- [NDcPP] Section 4
- [FWcPP] Section 3
- [IPS_EP] Section 2
- [VPN_EP] Section 3.

26. The statement of the Security Problem Definition in this ST is the superset of the threats, organizational security policies and assumptions from the collaborative Protection Profiles and Extended Packages. In creating this superset, some assumptions made in the [NDcPP] are restated as threats to be addressed by functionality specified in [FWcPP]. An additional assumption is introduced through compliance to [VPN_EP]. The addition of this assumption (from the point of view of conformance to [NDcPP], [FWcPP] and [IPS_EP]) is permissible for strict conformance[1] because A.CONNECTIONS does not mitigate a threat TOE nor fulfil an OSP meant to be addressed by security objectives for the TOE. Hence, this SPD statement is considered to be conformant to the collaborative Protection Profiles and Extended Packages claimed.

27. Similarly, the statement of security objectives in this ST is consistent with the statement of security objectives detailed in the collaborative Protection Profiles and Extended Packages to which this ST claims conformance, namely:

- The prose in [NDcPP] Section 3
- The prose in [FWcPP] Section 3
- [IPS_EP] Section 3
- [VPN_EP] Section 4.

[1] Exact Conformance, which doesn't say anything about modification of the SPD, is based on Strict Conformance

28. Again, the statement of the Security Objectives in this ST is the superset of the security objectives from the collaborative Protection Profiles and Extended Packages. In creating this superset, the objectives for the operational environment OE.NO_THRU_TRAFFIC_PROTECTION and OE.RESIDUAL_INFORMATION defined in [NDcPP] are omitted, as they are not relevant to this TOE; being addressed by objectives for the TOE introduced through conformance to [FWcPP]. Hence, the statement of security objectives in this ST is considered to be conformant to the collaborative Protection Profiles and Extended Packages claimed

29. The statement of requirements in this ST is consistent with the statement of requirements (functional and assurance) detailed in

- [NDcPP] Sections 6 & 7
- [FWcPP] Sections 5 & 6
- [IPS_EP] Section 4 & 6
- [VPN_EP] Section 5.

30. All Security Functional Requirements specified in [NDcPP] Section 6, together with the relevant selection-based requirements from Appendix B are included in this ST.

31. This statement of SFRs is augmented with SFRs specified in (at least one of) [NDcPP] Appendix A (Optional requirements), [FWcPP] Section 5 and Appendices A & B, [IPS_EP] Section 4 and [VPN_EP] Section 5 and Appendices A & B.

32. All extended requirements in this ST are taken from (at least one of) [NDcPP] Appendix C, [FWcPP] Appendix C, [IPS_EP] Section 4 and [VPN_EP] Section 5.

33. The Security Assurance Requirements specified in this ST are the superset of those defined in:

- [NDcPP] Section 7
- [FWcPP] Sections 6
- [IPS_EP] Section 6
- [VPN_EP] Section 5.4.

34. The distributed TOE deployment aspects described in [NDcPP] are not applicable as this TOE is satisfied by each model of the TOE in isolation.

2.3 Technical Decisions

35. In line with Labgram #105, this section identifies all NIAP Technical Decisions that are applicable to this TOE:

ITEM	TITLE	REFERENCE	PUBLICATION DATE	Relevant to ST
TD0453	NIT Technical Decision for Clarify authentication methods SSH clients can use to authenticate SSH servers	FCS_SSHC_EXT.1.9, ND SD v2.1	2019.09.16	No
TD0452	NIT Technical Decision for FCS_(D)TLSC_EXT.X.2 IP addresses in reference identifiers	FAU_GEN.1, ND SD v2.1	2019.09.16	No

ITEM	TITLE	REFERENCE	PUBLICATION DATE	Relevant to ST
TD0451	NIT Technical Decision for ITT Comm UUID Reference Identifier	FCS_TLSS_EXT.1.2 and FCS_TLSS_EXT.2.2	2019.09.16	No
TD0450	NIT Technical Decision for RSA-based ciphers and the Server Key Exchange message	FCS_TLSS_EXT.*.3, FCS_DTLSS_EXT.*.4, ND SD v2.1	2019.09.16	No
TD0449	NIT Technical Decision for Identification of usage of cryptographic schemes	FCS_CKM.2, ND SD v2.1	2019.09.16	Yes
TD0448	NIT Technical Decision for Documenting Diffie-Hellman 14 groups	FCS_CKM.2	2019.09.16	Yes
TD0447	NIT Technical Decision for Using 'diffie-hellman-group-exchange-sha256' in FCS_SSHC/S_EXT.1.7	FCS_SSHC_EXT.1.7, FCS_SSHS_EXT.1.7	2019.09.16	Yes
TD0436	IPsec protocol ESP algorithms	FCS_IPSEC_EXT.1.4	2019.07.19	Yes
TD0425	NIT Technical Decision for Cut-and-paste Error for Guidance AA	FTA_SSL.3	2019.05.31	Yes
TD0424	NIT Technical Decision for NDcPP v2.1 Clarification - FCS_SSHC/S_EXT1.5	FCS_SSHC_EXT.1.5, FCS_SSHS_EXT.1.5	2019.05.31	Yes
TD0423	NIT Technical Decision for Clarification about application of RfI#201726rev2	FTP_ITC.1, FTP_TRP.1/Admin, or FPT_ITT.1	2019.05.31	Yes
TD0412	NIT Technical Decision for FCS_SSHS_EXT.1.5 SFR and AA discrepancy	FCS_SSHS_EXT.1.5	2019.03.22	Yes
TD0411	NIT Technical Decision for FCS_SSHC_EXT.1.5, Test 1 - Server and client side seem to be confused	FCS_SSHC_EXT.1.5	2019.03.22	No
TD0410	NIT technical decision for Redundant assurance activities associated with FAU_GEN.1	FAU_GEN.1	2019.03.22	Yes
TD0409	NIT decision for Applicability of FIA_AFL.1 to key-based SSH authentication	FIA_AFL.1	2019.03.22	Yes
TD0408	NIT Technical Decision for local vs. remote administrator accounts	FIA_AFL.1, FIA_UAU_EXT.2, FMT_SMF.1	2019.03.22	Yes
TD0407	NIT Technical Decision for handling Certification of Cloud Deployments	N/A	2019.03.22	No
TD0402	NIT Technical Decision for RSA-based FCS_CKM.2 Selection	FCS_CKM.2	2019.02.24	No
TD0401	NIT Technical Decision for Reliance on external servers to meet SFRs	FTP_ITC.1	2019.02.24	Yes

ITEM	TITLE	REFERENCE	PUBLICATION DATE	Relevant to ST
TD0400	NIT Technical Decision for FCS_CKM.2 and elliptic curve-based key establishment	FCS_CKM.1, FCS_CKM.2	2019.02.24	Yes
TD0399	NIT Technical Decision for Manual installation of CRL (FIA_X509_EXT.2)	FIA_X509_EXT.2	2019.02.24	Yes
TD0398	NIT Technical Decision for FCS_SSH*EXT.1.1 RFCs for AES-CTR	FCS_SSHC_EXT.1.1, FCS_SSHS_EXT.1.1	2019.02.24	Yes
TD0397	NIT Technical Decision for Fixing AES-CTR Mode Tests	FCS_COP.1/ DataEncryption	2019.02.24	Yes
TD0396	NIT Technical Decision for FCS_TLSC_EXT.1.1, Test 2	FCS_DTLSC_EXT.1.1, FCS_DTLSC_EXT.2.1, FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.2.1	2019.02.24	No
TD0395	NIT Technical Decision for Different Handling of TLS1.1 and TLS1.2	FCS_TLSS_EXT.2.4, FCS_TLSS_EXT.2.5	2019.02.24	No
TD0394	NIT Technical Decision for Audit of Management Activities related to Cryptographic Keys	FAU_GEN.1	2019.02.24	Yes
TD0356	OE.CONNECTIONS added to VPN GW v2.1	N/A	2018.09.20	Yes
TD0343	NIT Technical Decision for Updating FCS_IPSEC_EXT.1.14 Tests	FCS_IPSEC_EXT.1.14	2018.08.02	Yes
TD0340	NIT Technical Decision for Handling of the basicConstraints extension in CA and leaf certificates	FIA_X509_EXT.1.1/Rev, item 3	2018.08.02	Yes
TD0339	NIT Technical Decision for Making password-based authentication optional in FCS_SSHS_EXT.1.2	FCS_SSHS_EXT.1.2	2018.08.02	Yes
TD0337	NIT Technical Decision for Selections in FCS_SSH*_EXT.1.6	FCS_SSHS_EXT.1	2018.08.02	Yes
TD0335	NIT Technical Decision for FCS_DTLS Mandatory Cipher Suites	FCS_DTLSC_EXT.1.1, FCS_DTLSC_EXT.2.1, FCS_DTLSS_EXT.1.1, FCS_DTLSS_EXT.2.1, FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.2.1, FCS_TLSS_EXT.1.1, FCS_TLSS_EXT.2.1	2018.08.01	No
TD0333	NIT Technical Decision for Applicability of FIA_X509_EXT.3	FIA_X509_EXT.3	2018.08.01	Yes
TD0329	IPSEC X.509 Authentication Requirements	FIA_X509_EXT.4, FCS_IPSEC_EXT.1.14 EP_VPN_GW_V2.1	2018.05.31	Yes
TD0325	Inline mode for Signature-based IPS policies	IPS_SBD_EXT.1.5	2018.05.21	Yes
TD0321	Protection of NTP communications	FTP_ITC.1, FPT_STM_EXT.1	2018.05.21	Yes

ITEM	TITLE	REFERENCE	PUBLICATION DATE	Relevant to ST
TD0319	Updates to FMT_SMF.1 in VPN Gateway EP	FMT_SMF.1 EP_VPN_GW_V2.1	2018.04.23	Yes
TD0317	FMT_MOF.1/Services and FMT_MTD.1/CryptoKeys	FMT_MOF.1/Services, FMT_MTD.1/CryptoKeys EP_VPN_GW_V2.1	2018.04.23	Yes
TD0316	Update to FPT_TST_EXT.2.1	FPT_TST_EXT.2.1, FPT_TST_EXT.3.1 EP_VPN_GW_V2.1	2018.04.20	Yes
TD0307	Modification of FTP_ITC_EXT.1.1	FTP_ITC_EXT.1.1, FTP_ITC.1.1 EP_VPN_GW_V2.1	2018.04.18	Yes
TD0291	NIT technical decision for DH14 and FCS_CKM.1	FCS_CKM.1.1	2018.02.03	Yes
TD0259	NIT Technical Decision for Support for X509 ssh rsa authentication IAW RFC 6187	FCS_SSHC_EXT.1.5, FCS_SSHS_EXT.1.5	2017.11.13	No
TD0248	FAU_GEN.1 Guidance Activity	EP_VPN_GW_V2.1, FAU_GEN.1	2017.10.20	n/a to ST
TD0242	FPF_RUL_EXT.1.7, Test 3 - Logging Dropped Packets	EP_VPN_GW_V2.1, FPF_RUL_EXT.1.7	2017.11.08	n/a to ST
TD0209	Additional DH Group added as selection for IKE Protocols	EP_VPN_GW_V2.1, FCS_IPSEC_EXT.1.11	2017.06.09	Yes
TD0179	Management Capabilities in VPN GW EP 2.1	EP_VPN_GW_V2.1, FMT_SMF.1.1	2017.04.11	Yes

Table 3 Applicable NIAP Technical Decisions

36. All other NIAP Technical Decisions fall into one of the following categories and hence are not applicable to this TOE:

- Relates to earlier version of cPPs/EPs claimed for this TOE. This TD has been superseded by cPPs/EPs (and associated SDs) released after this TD
- Relates to cPP/EP that is not claimed for this TOE

3 Security Problem Definition

37. As this TOE is not distributed, none of the threats/assumptions/OSPs relating to distributed TOEs are specified for this TOE.

3.1 Threats

38. The following threats for this TOE are as defined in [NDcPP] Section 4.1, which are also stated in [FWcPP], with editorial and terminology changes to reflect focus on firewall rather than general purpose network devices. Namely:

- T.UNAUTHORIZED_ADMINISTRATOR_ACCESS

  Threat agents may attempt to gain Administrator access to the network device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between network devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.

- T.WEAK_CRYPTOGRAPHY

  Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.

- T.UNTRUSTED_COMMUNICATION_CHANNELS

  Threat agents may attempt to target network devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the network device itself.

- T.WEAK_AUTHENTICATION_ENDPOINTS

  Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints – e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the network device itself could be compromised.

- T.UPDATE_COMPROMISE

  Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.

- T.UNDETECTED_ACTIVITY

  Threat agents may attempt to access, change, and/or modify the security functionality of the network device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.

- T.SECURITY_FUNCTIONALITY_COMPROMISE

  Threat agents may compromise credentials and device data enabling continued access to the network device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker.

- T.PASSWORD_CRACKING

  Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic, and may allow them to take advantage of any trust relationships with other network devices.

- T.SECURITY_FUNCTIONALITY_FAILURE

  A component of the network device may fail during start-up or during operations causing a compromise or failure in the security functionality of the network device, leaving the device susceptible to attackers.

39. The following threats additional threats specified in [FWcPP], [IPS_EP] and [VPN_EP] are also detailed for this TOE:

- T.NETWORK_DISCLOSURE

  An attacker may attempt to "map" a subnet to determine the machines that reside on the network, and obtaining the IP addresses of machines, as well as the services (ports) those machines are offering. This information could be used to mount attacks to those machines via the services that are exported.

- T. NETWORK_ACCESS

  With knowledge of the services that are exported by machines on a subnet, an attacker may attempt to exploit those services by mounting attacks against those services.

- T.NETWORK_MISUSE

  An attacker may attempt to use services that are exported by machines in a way that is unintended by a site's security policies. For example, an attacker might be able to use a service to "anonymize" the attacker's machine as they mount attacks against others.

40. The following threat specified in [FWcPP] only is also detailed for this TOE:

- T.MALICIOUS_TRAFFIC

  A stateful traffic filtering firewall also provides protections against malicious or malformed packets. It will protect against attacks like modification of connection state information and replay attacks. These attacks could cause the firewall, or the devices it protects, to grant unauthorized access or even create a Denial of Service.

41. The following threat specified in [IPS_EP] only is detailed for this TOE:

- T.NETWORK_DOS

  Attacks against services inside a protected network, or indirectly by virtue of access to malicious agents from within a protected network, might lead to denial of services otherwise available within a protected network. Resource exhaustion may occur in the event of co-ordinate service request flooding from a small number of sources.

42. The following threat specified in [VPN_EP] only is detailed for this TOE:

- T.DATA_INTEGRITY

  Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to modify the data without authorization. If known malicious external devices are able to communicate with devices on the protected network or if devices on the protected network can establish communications with those external devices then the data contained within the communications may be susceptible to a loss of integrity.

- T.HIJACKED_SESSION

  There may be an instance where a remote client's session is hijacked due to session activity. This could be accomplished because a user has walked away from the machine that was used to establish the session.

- T.REPLAY_ATTACK

  If an unauthorized individual successfully gains access to the system, the adversary may have the opportunity to conduct a "replay" attack. This method of attack allows the individual to capture packets traversing throughout the network and send the packets at a later time, possibly unknown by the intended receiver. Traffic is subject to replay if it meets the following conditions:

  - Cleartext: an attacker with the ability to view unencrypted traffic can identify an appropriate segment of the communications to replay as well in order to cause the desired outcome.
  - No integrity: alongside cleartext traffic, an attacker can make arbitrary modifications to captured traffic and replay it to cause the desired outcome if the recipient has no means to detect these modifications.

- T.UNAUTHORIZED_CONNECTION

  While a VPN client may have the necessary credentials (e.g., certificate, pre-shared key) to connect to a VPN gateway, there may be instances where the remote client, or the machine the client is operating on, has been compromised and attempts to make unauthorized connections.

- T.UNPROTECTED_TRAFFIC

  A remote machine's network traffic may be exposed to a hostile network. A user may be required to use a hostile (or unknown) network to send network traffic without being able to route the traffic appropriately.

43. No threats are identified for this TOE in addition to those specified in the collaborative Protection Profiles and Extended Packages.

3.2 Assumptions

44. The assumptions made for this TOE are as defined in [NDcPP] Section 4.2 and [FWcPP] Section 3.2 (with appropriate editorial and terminology differences to reflect general network device vs. firewall), namely:

- A.PHYSICAL_PROTECTION
- The network device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security and/or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP will not include

any requirements on physical tamper protection or other physical attack mitigations. The cPP will not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device.

- A.LIMITED_FUNCTIONALITY
- The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality).
- A.TRUSTED_ADMINSTRATOR

  The Security Administrator(s) for the network device are assumed to be trusted and to act in the best interest of security for the organization. This includes being appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The network device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device.

  For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
- A.REGULAR_UPDATES
- The network device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
- A.ADMIN_CREDENTIALS_SECURE
- The Administrator's credentials (private key) used to access the network device are protected by the platform on which they reside.

45. The following assumption A.RESIDUAL_INFORMATION and A.NO_THRU_TRAFFIC_PROTECTION defined in [NDcPP] are not relevant to this TOE as they are addressed by additional requirements introduced through conformance to [FWcPP].

- A.CONNECTIONS

  It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

46. The assumption A.CONNECTIONS is introduced through compliance to [VPN_EP] and [IPS_EP]. It is typically understood that an ST claiming exact compliance to a Protection Profile cannot introduce assumptions. However, that is on the understanding this limits applicability of the security functional requirements for the TOE, whereas this assumption is a clarification of how the manner in which the TOE is to be connected to distinct networks.

47. No assumptions are identified for this TOE in addition to those specified in the collaborative Protection Profiles and Extended Packages.

3.3 Organizational Security Policies

48. The OSP applied for this TOE is as defined in [NDcPP] Section 4.3 and [FWcPP] Section 3.3. The OSP P.ANALYZE applied for this TOE is as defined in [NDcPP] Section A.1.3. No additional OSPs are identified and no modification to the statement of OSPs is made for this TOE.

- 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.

- P.ANALYZE

  Analytical processes and information to derive conclusions about potential intrusions must be applied to IPS data and appropriate response actions taken.

4 Security Objectives

49. As this TOE is not distributed, none of the objectives relating to distributed TOEs are specified for this TOE.

4.1 Security Objectives for the TOE

50. The security objectives for the TOE are trivially determined through the inverse of the statement of threats presented in [NDcPP] Section 4.1 and [FWcPP] Section 3.1.

51. These are augmented by the statement of security objectives for the TOE in relation to the IPS capabilities as detailed in [IPS_EP] Section 3, namely:

- O.SYSTEM_MONITORING

  The IPS must collect and store information about all events that may indicate an IPS policy violation related to misuse, inappropriate access, or malicious activity on monitored networks.

- O.IPS_ANALYZE

  The IPS must apply analytical processes to network traffic data collected from monitored networks and derive conclusions about potential intrusions or network traffic policy violations.

- O.IPS_ANALYZE

  The IPS must respond appropriately to its analytical conclusions about IPS policy violations.

- O.TOE_ADMINISTRATION – as also defined by the inverse of the threats defined in [NDcPP] Section 4.1 and [FWcPP] Section 3.1

  The IPS will provide a method for authorized administrator to configure the TSF.

- O.TRUSTED_COMMUNICATIONS– as also defined by the inverse of the threats defined in [NDcPP] Section 4.1 and [FWcPP] Section 3.1

  The IPS will ensure that communications between distributed components of the TOE are not subject to unauthorized modification or disclosure.

52. These are further augmented by the statement of security objectives for the TOE in relation to the IPS capabilities as detailed in [VPN_EP] Section 4.1, namely:

- O.ADDRESS_FILTERING

  To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption or denial of services, and network-based reconnaissance, compliant TOE's will implement Packet Filtering capability. That capability will restrict the flow of network traffic between protected networks and other attached networks based on network addresses of the network nodes originating (source) and/or receiving (destination) applicable network traffic as well as on established connection information.

- O.ASSIGNED_PRIVATE_ADDRESS

  There are instances where a remote client desires secure communication with a gateway that is trusted. While a user may be connected via an untrusted network, it should still be possible to ensure that it can communicate with a known entity that controls the routing of the client's network packets. This can be accomplished by the VPN headend assigning an IP address that the gateway controls, as well as providing a routing point for the client's network traffic.

- O.AUTHENTICATION – as also defined by the inverse of the threats defined in [NDcPP] Section 4.1 and [FWcPP] Section 3.1

  To further address the issues associated with unauthorized disclosure of information, a compliant TOE's authentication ability (IPSec) will allow a VPN peer to establish VPN connectivity with another VPN peer. VPN endpoints authenticate each other to ensure they are communicating with an authorized external IT entity.

- O.CLIENT_ESTABLISHMENT_CONSTRAINTS

  To address the concern that a remote client may be compromised and attempt to establish connections with the headend VPN gateway outside of "normal" operations, this objective specifies conditions under which a remote client may establish connections. The administrator may configure the headend VPN gateway to accept a client's request for a connection based on attributes the administrator feels are appropriate.

- O.CRYPTOGRAPHIC_FUNCTIONS – as also defined by the inverse of the threats defined in [NDcPP] Section 4.1 and [FWcPP] Section 3.1

  To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption of services, and network-based reconnaissance, compliant TOEs will implement a cryptographic capabilities. These capabilities are intended to maintain confidentiality and allow for detection and modification of data that is transmitted outside of the TOE.

- O.FAIL_SECURE

  There may be instances where the TOE's hardware malfunctions or the integrity of the TOE's software is compromised, the latter being due to malicious or non-malicious intent. To address the concern of the TOE operating outside of its hardware or software specification, the TOE will shut down upon discovery of a problem reported via the self-test mechanism and provide signature-based validation of updates to the TSF.

- O.PORT_FILTERING

  To further address the issues associated with unauthorized disclosure of information, etc., a compliant TOE's port filtering capability will restrict the flow of network traffic between protected networks and other attached networks based on the originating (source) and/or receiving (destination) port (or service) identified in the network traffic as well as on established connection information.

- O.REMOTE_SESSION_TERMINATION

  A remote client's session can become vulnerable when there is a lack of activity. This is primarily due to a user walking away from a device that has a remote connection established. While some devices have a "lock screen" or logout capability, they cannot always assumed to be configured or available. To address this concern, a session termination capability is necessary during an administrator specified time period.

- O.SYSTEM_MONITORING – as also defined by the inverse of the threats defined in [NDcPP] Section 4.1 and [FWcPP] Section 3.1

  To address the issues of administrators being able to monitor the operations of the VPN gateway, it is necessary to provide a capability to monitor system activity. Compliant TOEs will implement the ability to log the flow of network traffic. Specifically, the TOE will provide the means for administrators to configure packet filtering rules to 'log' when network traffic is found to match the configured rule. As a result, matching a rule configured to 'log' will result in informative event logs whenever a match occurs. In addition, the establishment of security associations (SAs) is auditable, not only between peer VPN gateways, but also with certification authorities (CAs).

- O.TOE_ADMINISTRATION – as also defined by the inverse of the threats defined in [NDcPP] Section 4.1 and [FWcPP] Section 3.1

  Compliant TOEs will provide the functions necessary for an administrator to configure the packet filtering rules, as well as the cryptographic aspects of the IPsec protocol that are enforced by the TOE.

4.2 Security Objectives for the Operational Environment

53. The statement of security objectives for the operational environment of this TOE is as defined in [NDcPP] Section 5.1, [FWcPP] Section 4.1 (with appropriate editorial and terminology differences to reflect general network device vs. firewall), and [IPS_EP] Section A.2.2 namely:

- OE.PHYSICAL

  Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

- 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.TRUSTED_ADMIN

  Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

  For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted. [2]

- OE.UPDATES

  The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

- OE.ADMIN_CREDENTIALS_SECURE

  The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

- OE.CONNECTIONS[3]

  TOE administrators will ensure that the TOE is installed in a manner that will allow the TOE to effectively enforce its policies on network traffic of monitored networks.

54. The security objectives for the operational environment OE.NO_THRU_TRAFFIC_PROTECTION and OE.RESIDUAL_INFORMATION defined in [NDcPP] are not relevant to this TOE as they are

[2] As specified in Network Device Interpretation #201712rev 2, dated 29 Jan 2018, "NITDecisionRfI201712rev2"
[3] In accordance with TD0356

addressed by objectives for the TOE introduced through conformance to [FWcPP]. No additional security objectives for the operational environment are identified for this ST.

4.3 Security Objectives rationale

55. As these objectives for the TOE and operational environment are the same as those specified in [NDcPP], [FWcPP], [VPN_EP]and [IPS_EP], the rationales provided in the prose of the following are wholly applicable to this security target as the statements of threats, assumptions, OSPs and security objectives provided in this security target are the same as those defined in the collaborative Protection Profiles and Extended Packages to which this ST claims conformance

- [NDcPP] section 4
- [FWcPP], section 3
- [IPS_EP] section 3 & Annex A.2
- [VPN_EP] and section 5 and Annex A.

5 Security Functional Requirements

56. All security functional requirements are taken from the [NDcPP] and [FWcPP] collaborative Protection Profiles and from the [IPS_EP] and [VPN_EP] Extended Packages. The Security Functional requirements are primarily structured according to [NDcPP], with requirements and operations from [FWcPP], [IPS_EP] and [VPN_EP] inserted as appropriate. The SFRs are presented in accordance with the conventions described in [NDcPP] Section 6.1, and section 1.4 of this document.

57. Note: as this TOE is not distributed, none of the security functional requirements relating to distributed TOEs are specified for this TOE.

5.1 Security Audit (FAU)

5.1.1 Security Audit Data generation (FAU_GEN)

5.1.1.1 FAU_GEN.1 Audit data generation

FAU_GEN.1/ND Network Device Audit Data Generation

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

a) Start-up and shut-down of the audit functions;
b) All auditable events for the <u>not specified</u> level of audit; and
c) *All administrative actions comprising:*
   - *Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).*
   - *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
   - *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
   - *Resetting passwords (name of related user account shall be logged).*
   - *[<u>Starting and stopping services</u>];*

d) *Specifically defined auditable events listed in Table 4.*

ST Application Note:
The "Services" referenced in the above requirement relate to the trusted communication channel to the external syslog server (netconf over SSH) and the trusted path for remote administrative sessions (SSH, which can be tunneled over IPsec).

FAU_GEN.1.2/ND 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 ST, *information specified in column three of Table 4*.

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG_EXT.1	None	None
FAU_STG.1	None	None

FCS_CKM.1	None	None
FCS_CKM.2	None	None
FCS_CKM.4	None	None
FCS_COP.1/DataEncryption	None	None
FCS_COP.1/SigGen	None	None
FCS_COP.1/Hash	None	None
FCS_COP.1/KeyedHash	None	None
FCS_RBG_EXT.1	None	None
FDP_RIP.2	None	None
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None	None
FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None	None
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None
FMT_MTD.1/CoreData	All management activities of TSF data	None
FMT_SMF.1	None	None
FMT_SMR.2	None	None
FPT_SKP_EXT.1	None	None
FPT_APW_EXT.1	None	None
FPT_TST_EXT.1	None	None
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FPT_STM.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process.	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FTA_SSL_EXT.1	The termination of a local interactive session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None
FTA_SSL.4	The termination of an interactive session.	None
FTA_TAB.1	None	None
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1/Admin	Initiation of the trusted path.	None.

	Termination of the trusted path. Failure of the trusted path functions.	
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate	Reason for failure
FIA_X509_EXT.2	None	None
FIA_X509_EXT.3	None	None
FMT_MOF.1/Functions	Modification of the behaviour of the transmission of audit data to an external IT entity, the handling of audit data, the audit functionality when Local Audit Storage Space is full.	None.
FMT_MOF.1/Services	Starting and stopping of services.	None
FMT_MTD.1/CryptoKeys	Management of cryptographic keys.	None.
FFW_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport Layer Protocol TOE Interface
	Indication of packets dropped due to too much network traffic	TOE interface that is unable to process packets Identifier of rule causing packet drop
FFW_RUL_EXT.2	None	None
FCS_IPSEC_EXT.1	Session Establishment with peer	Entire packet contents of packets transmitted/received during session establishment
FIA_X509_EXT.1	Session establishment with CA	Entire packet contents of packets transmitted/received during session establishment
FPF_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport Layer Protocol TOE Interface
	Indication of packets dropped due to too much network traffic	TOE interface that is unable to process packets

Table 4 FAU_GEN.1 Security Functional Requirements and Auditable Events

FAU_GEN.1/IPS IPS Audit Data Generation[4]

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

a) Start-up and shut-down of the IPS functions;
b) All **IPS** auditable events for the [not specified] level of audit; and
c) ~~*All administrative actions;*~~
d) [*All dissimilar IPS events;*
e) All dissimilar IPS reactions;
f) Totals of similar events occurring within a specified time period; and
g) *Totals of similar reactions occurring within a specified time period.*]

FAU_GEN.1.2/IPS The TSF shall record within each **IPS auditable event** record at least the following information:

a) Date and time of the event, type of event **and/or reaction**; ~~subject identity, and the outcome (success or failure) of the event;~~ and
b) For each **IPS** audit**able** event type, based on the auditable event definitions of the functional components included in the ~~PP/~~ST, [*Specifically defined auditable events listed in* ~~*information specified in column three of*~~ *Table 5*].

Requirement	IPS Auditable Events	Additional Details
FMT_SMF.1/IPS	Modification of an IPS policy element.	Identifier or name of the modified IPS policy element (e.g. which signature, baseline, or known-good/known-bad list was modified.
IPS_ABD_EXT.1	Inspected traffic matches an anomaly-based IPS policy.	Source and destination IP addresses.
		The content of the header fields that were determined to match the policy.
		TOE interface that received the packet.
		Aspect of the anomaly-based IPS policy rule that triggered the event (e.g. throughput, time of day, frequency, etc.).
		Network-based action by the TOE (e.g. allowed, blocked, sent reset to source IP, sent blocking notification to firewall).
IPS_IPB_EXT.1	Inspected traffic matches a list of known-good or known-bad addresses applied to an IPS policy.	Source and destination IP addresses (and, if applicable, indication of whether the source and/or destination address matched the list).
		TOE interface that received the packet.

[4] Specified in [IPS_EP].

		Network-based action by the TOE (e.g. allowed, blocked, sent reset)
IPS_NTA_EXT.1	Modification of which IPS policies are active on a TOE interface. Enabling/disabling a TOE interface with IPS policies applied. Modification of which mode(s) is/are active on a TOE interface.	Identification of the TOE interface
		The IPS policy and interface mode (if applicable).
IPS_SBD_EXT.1	Inspected traffic matches a signature-based IPS with logging enabled.	Name or identifier of the matched signature.
		Source and destination IP addresses.
		The content of the header fields that were determined to match the signature.
		TOE interface that received the packet.
		Network-based action by the TOE (e.g. allowed, blocked, sent reset)

Table 5 Audit Events and Details from IPSEP

5.1.1.2 FAU_GEN.2 User identity association

FAU_GEN.2 User identity association[5]

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.

5.1.2 Security audit event storage (Extended – FAU_STG_EXT)

5.1.2.1 FAU_ STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1 Protected Audit Event Storage[6]

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 according to FTP_ITC.1.

ST Application Note
Transfer of the audit date to the external server is performed automatically (without further Security Administrator intervention) in the evaluated deployment.

FAU_STG_EXT.1.2 <u>TOE shall consist of a single standalone component that stores audit data locally</u>.

FAU_STG_EXT.1.3 The TSF shall [<u>overwrite previous audit records according to the following rule</u>: *[<u>oldest log is overwritten</u>]*] when the local storage space for audit data is full.

[5] Specified in [NDcPP], [FWcPP], [IPS_EP].
[6] Specified in [NDcPP], [FWcPP].

5.1.2.2 FAU_STG.1 Protected audit trail storage (Optional)

FAU_STG.1 Protected audit trail storage[7]

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

FAU_STG.1.2 The TSF shall be able to <u>prevent</u> unauthorised modifications to the stored audit records in the audit trail.

5.2 Cryptographic Support (FCS)

5.2.1 Cryptographic Key Management (FCS_CKM)

5.2.1.1 FCS_CKM.1 Cryptographic Key Generation (Refinement)

FCS_CKM.1/ND Cryptographic Key Generation/ND[8]

FCS_CKM.1.1/ND The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm: *[*

- *<u>RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;</u>*
- *<u>ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;</u>*
- *<u>FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3</u>*[9]

] ~~and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]~~.

5.2.1.2 FCS_CKM.1 Cryptographic Key Generation (Refinement)

FCS_CKM.1/IKE Cryptographic Key Generation (for IKE Peer Authentication)[10]

FCS_CKM.1.1/IKE The TSF shall generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with **a [**

- **<u>FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes;</u>**
- **<u>FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-256, P-384 and [no other curves]]</u>**

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

5.2.1.3 FCS_CKM.2 Cryptographic Key Establishment (Refinement)

FCS_CKM.2 Cryptographic Key Establishment

FCS_CKM.2.1 The TSF shall **perform** cryptographic **key establishment** in accordance with a specified cryptographic key **establishment** method: *[*

- *<u>Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";</u>*

---

[7] Specified in [NDcPP] and [FWcPP].
[8] Specified in [NDcPP], [FWcPP].
[9] As per TD0291 and Network Device Interpretations #201723rev2, in NITDecisionRfi201723rev2.pdf.
[10] Specified in [VPN_EP].

- *Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526, Section 3~~;~~*

] ~~that meets the following: [assignment: *list of standards*]~~.

5.2.1.4 FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

- *For plaintext keys in volatile storage, the destruction shall be executed by a [destruction of reference to the key directly followed by a request for garbage collection];*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [*
    - *instructs a part of the TSF to destroy the abstraction that represents the key]]*

that meets the following: *No Standard*.

5.2.2 Cryptographic Operation (FCS_COP)

5.2.2.1 FCS_COP.1 Cryptographic Operation

FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/Decryption)

FCS_COP.1.1/DataEncryption The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in [GCM, CBC, CTR] mode* and cryptographic key sizes *[128 bits, 192 bits, 256 bits]* that meet the following *AES as specified in ISO 18033-3, [CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772].*

FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)

FCS_COP.1.1/SigGen The TSF shall perform *cryptographic signature services (generation and verification)* in accordance with a specified cryptographic algorithm *[*

- *RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits, 4096 bits],*
- *Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits, 384 bits, 512 bits]*

] ~~and cryptographic key sizes [assignment: *cryptographic key sizes*]~~

that meet the following: *[*

- *For RSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,*
- *For ECDSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST curves” [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4*

].

FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1/Hash The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm *[SHA-1, SHA-256, SHA-384, SHA-512]* ~~and cryptographic key sizes [*assignment: cryptographic key sizes*]~~ and **message digest sizes [160, 256, 384, 512] bits** that meet the following: *ISO/IEC 10118-3:2004*.

FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm *[HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512]* and cryptographic key sizes [*160, 256, 384 and 512 bits*] **and message digest sizes [160, 256, 384, 512]** bits that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

5.2.3 Random Bit Generation (Extended – FCS_RBG_EXT)

5.2.3.1 FCS_RBG_EXT.1 Random Bit Generation

FCS_RBG_EXT.1 Random Bit Generation

FCS_RBG_EXT.1.1 The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [HMAC_DRBG (any)].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [*[2]*software-based noise source, *[1]* hardware-based noise source] with a minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011- Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

5.2.4 Cryptographic Protocols (Extended – FCS_ IPSEC_EXT & FCS_SSHS_EXT SSH Protocol

5.2.4.1 FCS_IPSEC_EXT.1 IPsec Protocol

FCS_IPSEC_EXT.1 IPsec Protocol

FCS_IPSEC_EXT.1.1 The TSF shall implement the IPsec architecture as specified in RFC 4301.

FCS_IPSEC_EXT.1.2 The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.

FCS_IPSEC_EXT.1.3 The TSF shall implement [tunnel mode].

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [AES-CBC-128, AES-CBC-256 (specified in RFC 3602) and [AES-CBC-192 (specified in RFC 3602)]] together with a Secure Hash Algorithm (SHA)-based HMAC [HMAC-SHA-256][11].

- ***FCS_IPSEC_EXT.1.5*** *The TSF shall implement the protocol [*IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [no other RFCs for extended sequence numbers], and [RFC 4868 for hash functions];
- IKEv2 as defined in RFC 5996 and [with no support for NAT traversal], and [RFC 4868 for hash functions]

]

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [IKEv1, IKEv2] protocol uses the cryptographic algorithms [AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 5282)].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

- IKEv1 Phase 1 SA lifetimes can be configured by a Security Administrator based on [
  - length of time, where the time values can be configured within [*0.2-24*] hours[12];

[11] The SFR is specified as per TD 0436.

[12] Length of time can be configured between 180 seconds and 86,400 seconds.

];
- IKEv2 SA lifetimes can be configured by a Security Administrator based on [
  - length of time, where the time values can be configured within [*0.2-24*] hours

  ]

].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

- IKEv1 Phase 2 SA lifetimes can be configured by a Security Administrator based on [
  - number of bytes;
  - length of time, where the time values can be configured within [*8*] hours;

  ];
- IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [
  - number of bytes;
  - length of time, where the time values can be configured within [*8*] hours;

  ]

].

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [*112 (for DH Group 14), 128 (for DH Groups 19 and 24) or 192 (for DH Group 20)*] bits.

FCS_IPSEC_EXT.1.10 The TSF shall generate nonces used in [IKEv1, IKEv2] exchanges of length [

at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash].

FCS_IPSEC_EXT.1.11 The TSF shall ensure that IKE protocols implement DH Group(s) [14 (2048-bit MODP), 19 (256-bit Random ECP), 20 (384-bit Random ECP), 24 (2048-bit MODP with 256-bit POS)][13].

FCS_IPSEC_EXT.1.12 The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 1, IKEv2 IKE_SA] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 2, IKEv2 CHILD_SA] connection.

FCS_IPSEC_EXT.1.13 The TSF shall ensure that all IKE protocols perform peer authentication using [RSA, ECDSA] that use X.509v3 certificates that conform to RFC 4945 and [Pre-shared Keys].

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are **Distinguished Name (DN)** [IP address, Fully Qualified Domain Name (FQDN), user FQDN] [14].

5.2.4.2 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) [4251, 4252, 4253, 4254, 5656, 6668][15].

---

[13] The SFR is specified as per [NDcPP] rather than the alternative presentation of the same requirements in [VPN_EP]. It also incorporates NIAP TD0209.

[14] The SFR is specified per [VPN_EP] as modified by TD0329 and TD0343.

[15] In accordance with TD0398

FCS_SSHS_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[16].

FCS_SSHS_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_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [*aes128-cbc, aes256-cbc, aes128-ctr, aes256-ctr*][17].

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [*ssh-rsa, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521*] as its public key algorithm(s) and rejects all other public key algorithms[18].

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses *[hmac-sha1, hmac-sha2-256, hmac-sha2-512] and [no other MAC algorithms*] as its MAC algorithm(s) and rejects all other MAC algorithm(s)[19].

FCS_SSHS_EXT.1.7 The TSF shall ensure that *[diffie-hellman-group14-sha1, ecdh-sha2-nistp256]* and *[ecdh-sha2-nistp384, edch-sha2-nistp521]* are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8 The TSF shall ensure that within SSH connections the same session keys are used for a threshold of no longer than one hour, and no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

5.3 Identification and Authentication (FIA)

5.3.1 Authentication Failure Management (FIA_AFL)

5.3.1.1 FIA_AFL.1 Authentication Failure Management (Refinement)

FIA_AFL.1 Authentication Failure Management[20]

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [*1 to 10*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.[21]

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

- *prevent the offending remote Administrator from successfully stablishing remote session using any authentication method that involves a password until [action= administrator has unlocked the account] is taken by an Administrator[22];*

or

- *prevent the offending remote Administrator from successfully authenticating until an Administrator defined time period has elapsed].*

[16] In accordance with TD0339
[17] In accordance with TD0337
[18] In accordance with TD0424
[19] In accordance with TD0337
[20] Specified in [NDcPP] and [VPN_EP], stated using the wording of [NDcPP].
[21] In accordance with TD0408
[22] In accordance with TD0408

ST Application Note
The Security Administrator can select to unlock the account of another administrator who has failed to authenticate, rather than require the administrator to wait until the delay of an administrator-configured time period has lapsed before another attempt can be made to authenticate.

5.3.2 Password Management (Extended – FIA_PMG_EXT)

5.3.2.1 FIA_PMG_EXT.1 Password Management

FIA_PMG_EXT.1 Password Management[23]

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: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [*and all other standard ASCII, extended ASCII and Unicode characters*]];
b) Minimum password length shall be configurable to **between** [*10*] and [*20*] **characters**.

5.3.3 User Identification and Authentication (Extended – FIA_UIA_EXT)

5.3.3.1 FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1 User Identification and Authentication[24]

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;
- [[*ICMP echo*]].

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.

5.3.4 User authentication (FIA_UAU) (Extended – FIA_UAU_EXT)

5.3.4.1 FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2 Password-based Authentication Mechanism[25]

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, and [no other authentication mechanism] to perform local administrative user authentication[26].

5.3.4.2 FIA_UAU.7 Protected Authentication Feedback

FIA_UAU.7 Protected Authentication Feedback[27]

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**.

[23] Specified in [NDcPP] and [FWcPP].
[24] Specified in [NDcPP] and [FWcPP].
[25] Specified in [NDcPP] and [FWcPP].
[26] In accordance with TD0408
[27] Specified in [NDcPP] and [FWcPP].

5.3.5 Authentication using X.509 certificates (Extended – FIA_X509_EXT)

5.3.5.1 FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1/Rev X.509 Certificate Validation

FIA_X509_EXT.1.1/Rev The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.[28]
- The TSF shall validate the revocation status of the certificate using [<u>a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3</u>]
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

FIA_X509_EXT.1.2/Rev The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.3.5.2 FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [<u>IPsec</u>], and [<u>no additional uses</u>].

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [<u>allow the Administrator to choose whether to accept the certificate in these cases, not accept the certificate</u>].

5.3.5.3 FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1 The TSF shall generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key and [<u>device-specific information, Common Name, Organization, Organizational Unit, Country</u>][29].

FIA_X509_EXT.3.2 The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

---

[28] In accordance with TD0340

[29] In accordance with TD0333

5.3.5.4 FIA_PSK_EXT.1 Pre-Shared Keys

5.3.5.5 FIA_PSK_EXT.1 Pre-Shared Keys [30]

FIA_PSK_EXT.1.1 The TSF shall be able to use pre-shared keys for IPsec and [no other protocols].

FIA_PSK_EXT.1.2 The TSF shall be able to accept text-based pre-shared keys that:

- are 22 characters and [*1 to 255bytes*];
- composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”).

ST Application Note:
The TOE accepts Unicode characters to specify text-based pre-shared keys. Unicode characters are encoded as UTF-8 and treated as multiple bytes – up to 4 bytes depending on the character. The maximum length limit for text-based pre-shared keys enforced by the TOE is 255 bytes. When a pre-shared key is only composed of ASCII characters this limit is equivalent to 255 characters.

FIA_PSK_EXT.1.3 The TSF shall condition the text-based pre-shared keys by using [SHA-1, [*conversion of the text string into an authentication value as per RFC 2409 for IKEv1 or RFC 4306 for IKEv2, using the pseudo-random function that is configured as the hash algorithm for the IKE exchanges*]].

FIA_PSK_EXT.1.4 The TSF shall be able to [accept] bit-based pre-shared keys.

5.4 Security Management (FMT)

5.4.1 Management of functions in TSF (FMT_MOF)

5.4.1.1 FMT_MOF.1/ManualUpdate Management of security functions behaviour

FMT_MOF.1/ManualUpdate Management of security functions behaviour

FMT_MOF.1.1/ManualUpdate The TSF shall restrict the ability to enable the functions *to perform manual updates* to *Security Administrators*.

5.4.1.2 FMT_MOF.1/Services Management of security functions behaviour

FMT_MOF.1/Services Management of security functions behaviour

FMT_MOF.1.1/Services The TSF shall restrict the ability to enable and disable the functions **and services** to *Security Administrators*.

5.4.1.3 FMT_MOF.1/Functions Management of security functions behaviour

FMT_MOF.1/Functions Management of security functions behaviour

FMT_MOF.1.1/Functions The TSF shall restrict the ability to [modify the behaviour of] the functions *[transmission of audit data to an external IT entity, handling of audit data]* to *Security Administrators*.

[30] Specified in [VPN_EP].

5.4.2 Management of TSF Data (FMT_MTD)

5.4.2.1 FMT_MTD.1/AdminAct Management of TSF Data

5.4.2.2 FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.1/CoreData Management of TSF Data

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

5.4.2.3 FMT_MTD.1/CryptoKeys Management of TSF data

FMT_MTD.1/CryptoKeys Management of TSF data[31]

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to *manage* the *cryptographic keys* **and certificates used for VPN operation** to *Security Administrators*.

5.4.3 Specification of Management Functions (FMT_SMF)

5.4.3.1 FMT_SMF.1 Specification of Management Functions

FMT_SMF.1/ND Specification of Management Functions for ND[32]

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using* ***digital signature and [no other]*** *capability prior to installing those updates;*
- ***Ability to configure the authentication failure parameters for FIA_AFL.1;***
- ***Ability to configure firewall rules****;*
- ***Ability to configure the cryptographic functionality;***
- ***Ability to configure the lifetime for IPsec SAs;***
- ***Ability to import X.509v3 certificates;***
- ***Ability to enable, disable, determine and modify the behavior of all the security functions of the TOE identified in ~~this EP~~ [VPN_EP] to the Administrator****;*
- ***Ability to configure all security management functions identified in [VPN_EP]~~other sections of this EP~~***

  [

  - *Ability to configure audit behaviour;*
  - *Ability to configure thresholds for SSH rekeying;*
  - *Ability to re-enable an Administrator account;*
  - *Ability to set the time which is used for time-stamps;*
  - *Ability to configure the reference identifier for the peer].*

]

FMT_SMF.1/IPS Specification of Management Functions for IPS[33]

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

[31] Specified in [VPN_EP] and TD0317
[32] Specified in [VPN_EP], TD0179 and TD0319
[33] Specified in [IPS_EP].

- *Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality*
- *Modify these parameters that define the network traffic to be collected and analyzed:*
    - *Source IP addresses (host address and network address)*
    - *Destination IP addresses (host address and network address)*
    - *Source port (TCP and UDP)*
    - *Destination port (TCP and UDP)*
    - *Protocol (IPv4 and IPv6)*
    - *ICMP type and code*
- *Update (import) signatures*
- *Create custom signatures*
- *Configure anomaly detection*
- *Enable and disable actions to be taken when signature or anomaly matches are detected*
- *Modify thresholds that trigger IPS reactions*
- *Modify the duration of traffic blocking actions*
- *Modify the known-good and known-bad lists (of IP addresses or address ranges)*
- *Configure the known-good and known-bad lists to override signature-based IPS policies*]

5.4.4 Security management roles (FMT_SMR)

5.4.4.1 FMT_SMR.2 Restrictions on security roles

FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security 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

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

are satisfied.

5.5 Protection of the TSF (FPT)

5.5.1 Protection of TSF Data (Extended – FPT_SKP_EXT)

5.5.1.1 FPT_SKP_EXT.1 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)

FPT_SKP_EXT.1 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)

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 (Extended – FPT_APW_EXT)

5.5.2.1 FPT_APW_EXT.1 Protection of Administrator Passwords

FPT_APW_EXT.1 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.

5.5.3 TSF testing (Extended – FPT_TST_EXT)

5.5.3.1 FPT_TST_EXT.1 TSF Testing (Extended)

FPT_TST_EXT.1 TSF testing

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [during initial start-up (on power on)] to demonstrate the correct operation of the TSF: [

- *Power on test,*
- *File integrity test,*
- *Crypto integrity test,*
- *Authentication test,*
- *Algorithm known answer tests*].

5.5.3.2 FPT_TST_EXT.2/VPN TSF Testing

FPT_TST_EXT.3 TSF Testing[34]

FPT_TST_EXT.3.1 The TSF shall provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the TSF-provided cryptographic service specified in FCS_COP.1**/SigGen**[35].

5.5.4 Trusted Update (FPT_TUD_EXT)

5.5.4.1 FPT_TUD_EXT.1 Trusted Update

FPT_TUD_EXT.1 Trusted update

FPT_TUD_EXT.1.1 The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [no other TOE firmware/software version].

FPT_TUD_EXT.1.2 The TSF shall provide *Security Administrators* the ability to manually initiate updates to TOE firmware/software and [no other update mechanism].

FPT_TUD_EXT.1.3 The TSF shall provide means to authenticate firmware/software updates to the TOE using a digital signature mechanism *and* [no other mechanisms] prior to installing those updates[36].

5.5.5 Time stamps (Extended – FPT_STM_EXT))

5.5.5.1 FPT_STM_EXT.1 Reliable Time Stamps

FPT_STM_EXT.1 Reliable Time Stamps

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

FPT_STM_EXT.1.2 The TSF shall [allow the Security Administrator to set the time].

[34] Specified in [VPN_EP].
[35] In accordance with TD0316
[36] The wording is taken from [VPN_EP].

5.5.6 Self-test Failures (FPT_FLS)

5.5.6.1 FPT_FLS.1/SelfTest Fail Secure

5.5.6.1 FPT_FLS.1/SelfTest Fail Secure [37]

FPT_FLS.1.1/SelfTest The TSF shall **shut down** when the following types of failures occur: [*failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests*].

5.6 TOE Access (FTA)

5.6.1 TSF-initiated Session Locking (Extended – FTA_SSL_EXT)

5.6.1.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1 TSF-initiated Session Locking[38]

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

- <u>terminate the session]</u>

after a Security Administrator-specified time period of inactivity.

5.6.2 Session locking and termination (FTA_SSL)

5.6.2.1 FTA_SSL.3 TSF-initiated Termination (Refinement)

FTA_SSL.3 TSF-initiated Termination[39]

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

5.6.2.2 FTA_SSL.4 User-initiated Termination (Refinement)

FTA_SSL.4 User-initiated Termination[40]

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

5.6.3 TOE access banners (FTA_TAB)

5.6.3.1 FTA_TAB.1 Default TOE Access Banners (Refinement)

FTA_TAB.1 Default TOE Access Banners[41]

FTA_TAB.1.1: 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.

[37] Specified in [VPN_EP].
[38] Specified in [NDcPP] and [FWcPP].
[39] Specified in [NDcPP] and [FWcPP].
[40] Specified in [NDcPP] and [FWcPP].
[41] Specified in [NDcPP] and [FWcPP].

5.7 Trusted path/channels (FTP)

5.7.1 Trusted Channel (FTP_ITC)

5.7.1.1 FTP_ITC.1 Inter-TSF trusted channel (Refinement)

FTP_ITC.1 Inter-TSF trusted channel[42]

FTP_ITC.1.1 Refinement: The TSF shall use **IPsec, and [SSH]** to provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, VPN communications, [no other capabilities]** 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[43].

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 [*VPN communications and streaming of syslog events*].

5.7.2 Trusted Path (FTP_TRP)

5.7.2.1 FTP_TRP.1/Admin Trusted Path (Refinement)

FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin The TSF shall **be capable of using [SSH, IPSEC] to** provide a communication path between itself and **authorized** 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 provides detection of modification of the channel data**.

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

FTP_TRP.1.3/Admin The TSF shall require the use of the trusted path *for initial Administrator authentication and all remote administration actions*.

5.8 User Data Protection (FDP)

5.8.1 Residual information protection (FDP_RIP)

5.8.1.1 FDP_RIP.2 Full Residual Information Protection

FDP_RIP.2 Full Residual Information Protection[44]

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.

[42] The wording of this element is taken from [VPN_EP] and TD0307.
[43] The wording of this element is taken from [VPN_EP].
[44] Specified in [FWcPP].

5.9 Packet Filtering (FPF)

5.9.1 Packet Filtering Rules (FPF_RUL_EXT)

5.9.1.1 FPF_RUL_EXT.1 Rules for Packet Filtering

FPF_RUL_EXT.1 Rules for Packet Filtering[45]

FPF_RUL_EXT.1.1 The TSF shall perform Packet Filtering on network packets processed by the TOE.

FPF_RUL_EXT.1.2 The TSF shall process the following network traffic protocols:

- Internet Protocol (IPv4)
- Internet Protocol version 6 (IPv6)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)

and be capable of inspecting network packet header fields defined by the following RFCs to the extent mandated in the other elements of this SFR

- RFC 791 (IPv4)
- RFC 2460 (IPv6)
- RFC 793 (TCP)
- RFC 768 (UDP).

FPF_RUL_EXT.1.3 The TSF shall allow the definition of Packet Filtering rules using the following network protocol fields:

- IPv4
  - Source address
  - Destination Address
  - Protocol
- IPv6
  - Source address
  - Destination Address
  - Next Header (Protocol)
- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port

FPF_RUL_EXT.1.4 The TSF shall allow the following operations to be associated with Packet Filtering rules: permit, discard, and log.

FPF_RUL_EXT.1.5 The TSF shall allow the Packet Filtering rules to be assigned to each distinct network interface.

FPF_RUL_EXT.1.6 The TSF shall process the applicable Packet Filtering rules (as determined in accordance with FPF_RUL_EXT.1.5) in the following order: Administrator-defined.

FPF_RUL_EXT.1.7 The TSF shall drop traffic if a matching rule is not identified.

[45] Specified in [VPN_EP].

5.10 Firewall (FFW)

5.10.1 Stateful Traffic Filter Firewall (FFW_RUL_EXT)

5.10.1.1 FFW_RUL_EXT.1 Stateful Traffic Filtering

FFW_RUL_EXT.1 Stateful Traffic Filtering

FFW_RUL_EXT.1.1 The TSF shall perform Stateful Traffic Filtering on network packets processed by the TOE.

FFW_RUL_EXT.1.2 The TSF shall allow the definition of Stateful Traffic Filtering rules using the following network protocol fields:

- ICMPv4
  - Type
  - Code
- ICMPv6
  - Type
  - Code
- IPv4
  - Source address
  - Destination Address
  - Transport Layer Protocol
- IPv6
  - Source address
  - Destination Address
  - Transport Layer Protocol
  - [no other field]
- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port
- and distinct interface.

FFW_RUL_EXT.1.3 The TSF shall allow the following operations to be associated with Stateful Traffic Filtering rules: permit or drop with the capability to log the operation.

FFW_RUL_EXT.1.4 The TSF shall allow the Stateful Traffic Filtering rules to be assigned to each distinct network interface.

FFW_RUL_EXT.1.5 The TSF shall

a) accept a network packet without further processing of Stateful Traffic Filtering rules if it matches an allowed established session for the following protocols: TCP, UDP, [ICMP] based on the following network packet attributes:
   1. TCP: source and destination addresses, source and destination ports, sequence number, Flags;
   2. UDP: source and destination addresses, source and destination ports;
   3. [ICMP: source and destination addresses, type, [code]].

b) Remove existing traffic flows from the set of established traffic flows based on the following: [session inactivity timeout, completion of the expected information flow].

FFW_RUL_EXT.1.6 The TSF shall enforce the following default Stateful Traffic Filtering rules on all network traffic:

a) The TSF shall drop and be capable of [logging] packets which are invalid fragments;
b) The TSF shall drop and be capable of [logging] fragmented packets which cannot be re-assembled completely;
c) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a broadcast network;
d) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a multicast network;
   The TSF shall drop and be capable of logging network packets where the source address of the network packet is defined as being a loopback address;
e) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address “reserved for future use” (i.e. 240.0.0.0/4) as specified in RFC 5735 for IPv4;
f) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is defined as an “unspecified address” or an address “reserved for future definition and use” (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6;
g) The TSF shall drop and be capable of logging network packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified; and
h) [no other rules].

FFW_RUL_EXT.1.7 The TSF shall be capable of dropping and logging according to the following rules:

a) The TSF shall drop and be capable of logging network packets where the source address of the network packet is equal to the address of the network interface where the network packet was received;
b) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is a link-local address;
c) The TSF shall drop and be capable of logging network packets where the source address of the network packet does not belong to the networks associated with the network interface where the network packet was received.

FFW_RUL_EXT.1.8 The TSF shall process the applicable Stateful Traffic Filtering rules in an administratively defined order.

FFW_RUL_EXT.1.9 The TSF shall deny packet flow if a matching rule is not identified.

FFW_RUL_EXT.1.10 The TSF shall be capable of limiting an administratively defined number of half-open TCP connections. In the event that the configured limit is reached, new connection attempts shall be dropped and the drop event shall be [logged].

5.10.1.2 FFW_RUL_EXT.2 Stateful Filtering of Dynamic Protocols

FFW_RUL_EXT.2 Stateful Filtering of Dynamic Protocols

FFW_RUL_EXT.2.1 The TSF shall dynamically define rules or establish sessions allowing network traffic to flow for the following network protocols [FTP].

5.11 Intrusion Prevention (IPS)

5.11.1 Network Traffic Analysis (IPS_NTA_EXT)

5.11.1.1 Network Traffic Analysis

IPS_NTA_EXT.1 Network Traffic Analysis

IPS_NTA_EXT.1.1 The TSF shall perform analysis of IP-based network traffic forwarded to the TOE's sensor interfaces, and detect violations of administratively-defined IPS policies.

IPS_NTA_EXT.1.2 The TSF shall process (be capable of inspecting) the following network traffic protocols:

- Internet Protocol (IPv4), RFC 791
- Internet Protocol version 6 (IPv6), RFC 2460
- Internet control message protocol version 4 (ICMPv4), RFC 792
- Internet control message protocol version 6 (ICMPv6), RFC 2463
- Transmission Control Protocol (TCP), RFC 793
- User Data Protocol (UDP), RFC 768

IPS_NTA_EXT.1.3 The TSF shall allow the signatures to be assigned to sensor interfaces configured for promiscuous mode, and to interfaces configured for inline mode, and support designation of one or more interfaces as 'management' for communication between the TOE and external entities without simultaneously being sensor interfaces.

- Promiscuous (listen-only) mode: [*none*];
- Inline (data pass-through) mode: [*Ethernet interfaces*];
- Management mode: [*Ethernet interfaces, out-of-band management Ethernet interfaces*];
    - o Session-reset-capable interfaces: [*Ethernet interfaces*];
    - o and no other interface types].

5.11.2 IPS IP Blocking (IPS_IPB_EXT)

5.11.2.1 IP Blocking

IPS_IPB_EXT.1 IP Blocking

IPS_IPB_EXT.1.1 The TSF shall support configuration and implementation of known-good and known-bad lists of [source, destination] IP addresses.

IPS_IPB_EXT.1.2 The TSF shall allow IPS Administrators and [no other roles] to configure the following IPS policy elements: [known-good list rules, known-bad list rules, IP addresses].

5.11.3 Signature-Based IPS Functionality (IPS_SBD_EXT)

5.11.3.1 Signature-Based IPS

IPS_SBD_EXT.1 Signature-Based IPS Functionality

IPS_SBD_EXT.1.1 The TSF shall support inspection of packet header contents and be able to inspect at least the following header fields:

- IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; IP Options; and [no other field].
- IPv6: Version; payload length; next header; hop limit; source address; destination address; routing header; and [traffic class, flow label].

- ICMP: Type; Code; Header Checksum; and [*rest of header (varies based on the ICMP type and code)*]].
- ICMPv6: Type; Code; and Header Checksum.
- TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.
- UDP: Source port; destination port; length; and UDP checksum.

IPS_SBD_EXT.1.2 The TSF shall support inspection of packet payload data and be able to inspect at least the following data elements to perform string-based pattern-matching:

- ICMPv4 data: characters beyond the first 4 bytes of the ICMP header.
- ICMPv6 data: characters beyond the first 4 bytes of the ICMP header.
- TCP data (characters beyond the 20 byte TCP header), with support for detection of:
  - i) FTP (file transfer) commands: help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.
  - ii) HTTP (web) commands and content: commands including GET and POST, and administrator-defined strings to match URLs/URIs, and web page content.
  - iii) SMTP (email) states: start state, SMTP commands state, mail header state, mail body state, abort state.
  - iv) [no other types of TCP payload inspection];
- UDP data: characters beyond the first 8 bytes of the UDP header;
- *[no other types of packet payload inspection];*

In addition, the TSF shall support stream reassembly or equivalent to detect malicious payload even if it is split across multiple non-fragmented packets.

IPS_SBD_EXT.1.3 The TSF shall be able to detect the following header-based signatures (using fields identified in IPS_SBD_EXT.1.1) at IPS sensor interfaces:

a) IP Attacks
   i. IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)
   ii. ii) IP source address equal to the IP destination (Land attack)
b) ICMP Attacks
   i. Fragmented ICMP Traffic (e.g. Nuke attack)
   ii. Large ICMP Traffic (Ping of Death attack)
c) TCP Attacks
   i. TCP NULL flags
   ii. TCP SYN+FIN flags
   iii. TCP FIN only flags
   iv. TCP SYN+RST flags
d) UDP Attacks
   i. UDP Bomb Attack
   ii. UDP Chargen DoS Attack

IPS_SBD_EXT.1.4 The TSF shall be able to detect all the following traffic-pattern detection signatures, and to have these signatures applied to IPS sensor interfaces:

a) Flooding a host (DoS attack)
   i. ICMP flooding (Smurf attack, and ping flood)
   ii. TCP flooding (e.g. SYN flood)
b) Flooding a network (DoS attack)
c) Protocol and port scanning
   i. IP protocol scanning
   ii. TCP port scanning

iii. UDP port scanning
iv. ICMP scanning

IPS_SBD_EXT.1.5 The TSF shall allow the following operations to be associated with signature-based IPS policies:

- In any mode, for any sensor interface: [
  - allow the traffic flow;
  - send a send a TCP reset to the source address of the offending traffic;
  - send a TCP reset to the destination address of the offending traffic]
- In inline mode:
  - block/drop the traffic flow;
  - and [allow all traffic flow] [46]

5.11.4 Anomaly-Based IPS Functionality (IPS_ABD_EXT)

5.11.4.1 *IPS_ABD_EXT.1 Anomaly-Based IPS Functionality*

IPS_ABD_EXT.1 Anomaly-Based IPS

IPS_ABD_EXT.1.1 The TSF shall support the definition of [anomaly ('unexpected') traffic patterns] including the specification of [

- throughput ([*bits per second*]);
- time of day;
- frequency;
- thresholds;
- [*no other methods*]]

and the following network protocol fields:

- *[IPv4: source address; destination address*
- *IPv6: source address; destination address*
- *TCP: source port; destination port*
- *UDP: source port; destination port]*

IPS_ABD_EXT.1.2 The TSF shall support the definition of anomaly activity through [manual configuration by administrators].

IPS_ABD_EXT.1.3 The TSF shall allow the following operations to be associated with anomaly-based IPS policies:

- In any mode, for any sensor interface: [
  - allow the traffic flow;
  - send a send a TCP reset to the source address of the offending traffic;
  - send a TCP reset to the destination address of the offending traffic]
- In inline mode:
  - allow the traffic flow
  - block/drop the traffic flow
- and [no other actions]

[46] Updated per NIAP TD0325

6 Security Assurance Requirements

58. The TOE security assurance requirements are taken from [NDcPP] , together with the refinements documented in [NDcPP] Section 7, as listed in Table 6 below.

Assurance Class	Assurance Component
Security Target (ASE)	Conformance claims (ASE_CCL.1)
	Extended components definition (ASE_ECD.1)
	ST introduction (ASE_INT.1)
	Security objectives for the operational environment (ASE_OBJ.1)
	Stated security requirements (ASE_REQ.1)
	Security Problem Definition (ASE_SPD.1)
Development (ADV)	TOE summary specification (ASE_TSS.1)
Guidance documents (AGD)	Basic functional specification (ADV_FSP.1)
	Operational user guidance (AGD_OPE.1)
Life cycle support (ALC)	Preparative procedures (AGD_PRE.1)
	Labelling of the TOE (ALC_CMC.1)
	TOE CM coverage (ALC_CMS.1)
Tests (ATE)	Independent testing – conformance (ATE_IND.1)
Vulnerability assessment (AVA)	Vulnerability survey (AVA_VAN.1)

Table 6 Security Assurance Requirements

7 TOE Summary Specification

7.1 Protected communications

59. Local console access is gained by connecting an RJ-45 cable between the console port on the appliance and a workstation with a serial connection client.

7.1.1 Algorithms and zeroization

60. All FIPS-approved cryptographic functions implemented by the secure (virtual) network appliance are implemented in the following three main libraries: Authentec (for IKE), Data plane (for IPSec), and OpenSSL (for SSH and the PKI daemon). All random number generation by the TOE is performed in accordance with NIST Special Publication 800-90 using HMAC_DRBG implemented in the OpenSSL library (FCS_RBG_EXT.1.1). Additionally, SHA (256,512) is implemented in the LibMD library which is used for password hashing by Junos' MGD daemon. **The network appliance is to be operated with FIPS mode enabled.**

61. The TOE evaluation provides a CAVP validation certificate for all FIPS-approved cryptographic functions implemented by the TOE. CAVP certificate details are provided in Table 7, below.

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	Certificate Number	SRX1500	SRX4100/4200	SRX4600
					Processor	Processor	Processor
SPU: Data Plane	Data plane – IPSec Daemon SRXPFE/QAT	FIPS 197, SP 800-38D	AES-GCM (128, 256), (192) (Encrypt, Decrypt, AEAD)	C1046	Intel(R) Xeon(R) CPU E3-1200 v2 with Intel Cave Creek QAT for crypto (FW/HW)	Intel(R) Xeon(R) CPU E5-2640 v4 with Intel Coleto Creek QAT (FW/HW)	Intel(R) Xeon(R) CPU E5-2658 v4 with Intel Coleto Creek QAT (HW/FW)
		FIPS 197, SP 800-38A	AES-CBC (128, 256), (192) (Encrypt, Decrypt)	C1046			
		FIPS 180-4	SHS: SHA (1, 256) Byte Oriented (Message Digest Generation)	C1046			
		FIPS 198-1	HMAC-SHA (1,256) Byte Oriented (Message Authentication)	C1046			
RE: Control Plane	Quicksec7– IKED Daemon	FIPS 197, SP 800-38A	AES-CBC (128, 256), (192) (Encrypt, Decrypt)	C1045	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
		FIPS 197, SP 800-38D	AES-GCM (128, 256) (Encrypt, Decrypt, AEAD)	C1045			
		FIPS 180-4	SHS: SHA (256, 384) Byte Oriented (Message Digest Generation)	C1045			
		FIPS 198-1	HMAC-SHA (256, 384)	C1045			

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	Certificate Number	SRX1500	SRX4100/4200	SRX4600
					Processor	Processor	Processor
RE: Control Plane	Quicksec7– All Daemons	SP 800-90A	DRBG (HMAC-SHA-256) (Random Bit Generation)	C1045	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
SPU: Control Plane	Quicksec7 – IKED Daemon	FIPS 186-4	RSA PKCS1_V1_5 (n=2048 (SHA 256), n=4096 (SHA 256)) (SigGen, SigVer)	C1045	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
		FIPS 186-4	ECDSA (P-256 w/ SHA-256) ECDSA (P-384 w/ SHA-384) (SigGen, SigVer)	C1045			
		SP 800-135	CVL IKEv1 KDF (SHA 256, 384)	C1045	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
			CVL IKEv2 KDF (SHA 256, 384)	C1045			
RE: Control Plan	OpenSSL – IKED Daemon	SP 800-56 A	CVL / KASVS FFC (DH Groups 24)	C1049	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
			CVL / KASVS ECC (ECDH Groups 19 and 20)	C1049			
RE: Control Plane	OpenSSL – SSHD Daemon and PKID Daemon	FIPS 197, SP800-38A	AES-CBC/CTR (128, 192, 256) (Encrypt, Decrypt)	C1049	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	Certificate Number	SRX1500	SRX4100/4200	SRX4600
					Processor	Processor	Processor
		FIPS 180-4	SHS: SHA (1, 256, 384) Byte Oriented (Message Digest Generation, KDF Primitive; SHA1 is used for dh-group14-sha1 SSH key-exchange)	C1049			
		FIPS 180-4	SHS: SHA-512 Byte Oriented (Message Digest Generation)	C1049			
		FIPS 198-1	HMAC-SHA (1, 256), (512) Byte Oriented (Message Authentication DRBG Primitive)	C1049			
		FIPS 186-4	RSA PKCS1_V1_5 (n=2048 (SHA-256), n=4096 (SHA-256)) (SigGen, SigVer)	C1049			
		FIPS 186-4	RSA X931 (n=2048 (SHA-256), n=4096 (SHA-256)) (KeyGen)	C1049			

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	Certificate Number	SRX1500	SRX4100/4200	SRX4600
					Processor	Processor	Processor
		FIPS 186-4	ECDSA [P-256 (SHA-256)] , [P-384 (SHA-384)], [P-521 (SHA-512)] (SigGen, SigVer, KeyGen)	C1049			
		SP 800-56A	KASVS ECC (P-256, P-384, P-521)	C1049			
	OpenSSL – All Daemons	SP 800-90A	DRBG (HMAC-SHA-2-256) (Random Bit Generation)	C1049			
RE: Control Plane	OpenSSH – SSHD Daemon	SP 800-135	CVL SSH v2 KDF (SHA-256, SHA-384) (Key Derivation)	C1050	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
RE: Control Plane	Libmd - MGD Daemon, Password Hashing	FIPS 180-4	SHS: SHA (256, 512) Byte Oriented (Message Digest Generation)	C1043	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
RE: Control Plane	Kernel – Veriexec	FIPS 180-4	SHS: SHA1, 256 Byte Oriented (Message Digest Generation)	C1044	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)
	Kernel – kernel-hmac drbg	SP800-90A	DRBG (HMAC-SHA-2-256) (RBG)	C1044	Intel(R) Xeon(R) CPU E3-1200 v2 (FW)	Intel(R) Xeon(R) CPU E5-2640 v4 (FW)	Intel(R) Xeon(R) CPU E5-2658 v4 (FW)

Table 7 CAVP Certificate Results for Cryptographic Services

62. The FIPS approved algorithms are applied when the FIPS operating mode is enabled[47]. The relevant FIPS knobs are specified in [ECG]. (***FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, FCS_COP.1/KeyedHash, FCS_RBG_EXT.1, FCS_CKM.1, FMT_SMF.1***)

63. Asymmetric keys are generated in accordance with NIST SP 800-56A and FIPS PUB 186-4 for IKE with IPSec. The TOE complies with section 5.6 of NIST SP 800-56A regarding asymmetric key pair generation. The TOE implements all of the "shall" and "should" requirements and none of the "shall not" or "should not" from FIPS PUB 186-4 Appendix B3 and B4. (***FCS_CKM.1/IKE***)

64. Asymmetric keys are also generated in accordance with FIPS PUB 186-4 Appendix B.3 for RSA Schemes and Appendix B.4 for ECC Schemes for SSH communications. The TOE implements Diffie-Hellman group 14, using the modulus and generator specified by Section 3 of RFC3526. (***FCS_CKM.2, FCS_CKM.1/ND***).

65. The following table relates cryptographic algorithms to the protocols by the TOE. The TOE acts as both sender and recipient for IPsec and only as the server for SSH in the supported protocols listed in Table 8:

Protocol	Key Exchange	Auth	Cipher	Integrity
IKEv1	Group 14 (modp 2048) Group 19 (P-256) Group 20 (P-384) Group 24 (modp 2048)	RSA 2048, RSA 4096 ECDSA P-256 ECDSA P-384 Pre-Shared Key	AES CBC 128 AES CBC 192 AES CBC 256	HMAC-SHA-256-128 HMAC-SHA-384-192
IKEv2	Group 14 (modp 2048) Group 19 (P-256) Group 20 (P-384) Group 24 (modp 2048)	RSA 2048, RSA 4096 ECDSA P-256 ECDSA P-384 Pre-Shared Key	AES CBC 128 AES CBC 192 AES CBC 256 AES GCM 128 AES GCM 192 AES GCM 256	HMAC-SHA-256-128 HMAC-SHA-384-192
IPsec ESP	IKEv1 with optional: Group 14 (modp 2048) Group 19 (P-256) Group 20 (P-384) Group 24 (modp 2048)	IKEv1	AES CBC 128 AES CBC 192 AES CBC 256	HMAC-SHA-256-128
	IKEv2 with optional: Group 14 (modp 2048) Group 19 (P-256) Group 20 (P-384) Group 24 (modp 2048)	IKEv2	AES CBC 128 AES CBC 192 AES CBC 256 AES GCM 128 AES GCM 192 AES GCM 256	HMAC-SHA-256-128
SSHv2	DH Group 14 (modp 2048) ECDH-sha2-nistp256 ECDH-sha2-nistp384 ECDH-sha2-nistp521	SSH-RSA ECDSA P-256 ECDSA P-384, ECDSA P-521	AES CTR 128 AES CTR 256 AES CBC 128 AES CBC 256	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512

Table 8 Supported Protocols

66. The integrity algorithm HMAC-SHA-1 uses key length 160 bits, block size 512 bits and output size 160 bits. HMAC-SHA-256 uses key length 256 bits, block size 512 bits and output size 256 bits.

[47] The knob “set system fips level 2” will enforce strict compliance to FIPS and enable restrictions on algorithms and keys sizes as required by FIPS requirements.

HMAC-SHA-384 uses key length 384 bits, block size 1024 bits and output size 384 bits. HMAC-SHA-512 uses key length 512 bits, block size 1024 bits and output size 512 bits.

67. Junos OS handles zeroization for all CSP, plaintext secret and private cryptographic keys according to Table 9 below. (***FCS_CKM.4***).

CSP	Description	Method of storage	Storage location	Zeroization Method
SSH Private Host Key	The first time SSH is configured the set of Host keys is generated. Used to identify the host. ecdsa-sha2-nistp256 (ECDSA P-256) and/or ssh-rsa (RSA 2048)	Plaintext	File format on Virtual Disk (mapped to SDD)	When the appliance is recommissioned, the config files (including CSP files such as SSH keys) are removed using the Linux `shred` command to wipe the underlying persistent storage media.
	Loaded into memory to complete session establishment	Plaintext	Memory	Memory free() operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)
SSH Session Key	Session keys used with SSH, AES 128, 256, hmac-sha-1, hmac-sha2-256 or hmac-sha2-512 key (160, 256 or 512), DH Private Key (2048 or elliptic curve 256/384/521-bits)	Plaintext	Memory	Memory free() operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)
User Password	Plaintext value as entered by user	Plaintext as entered	Processed in Memory	Memory free() operation is performed by Junos (When released by the Junos VM, the KVM hypervisor erases the released memory before it is placed in the free pool).
		Hashed when stored (HMAC-sha1, sha256, sha512)	Stored on Virtual disk (mapped to SDD)	When the appliance is recommissioned, the config files (including the obfuscated password) are removed using the “request system zeroise hypervisor” option.
RNG State	Internal state and seed key of RNG	Plaintext	Memory	Handled by kernel, overwritten with zero’s at reboot.
IKE Private Host key	Private authentication key used in IKE. RSA 2048, ECDSA P-256, ECDSA P-384	Plaintext	Virtual disk (mapped to SDD)/Memory	‘clear security IKE security-association’ command or reboot the box. Private keys stored in flash are not zeroized unless an explicit “request system zeroize” is executed.
IKE-SKEYID	IKE master secret used to derive IKE and IPsec ESP session keys	Plaintext	Memory	‘clear security IKE security-association’ command or reboot the box

CSP	Description	Method of storage	Storage location	Zeroization Method
IKE Session Keys	IKE session key. AES, HMAC	Plaintext	Memory	'clear security IKE security-association' command or reboot the box
ESP Session Key	ESP session keys. AES, HMAC	Plaintext	Memory	'clear security ipsec security-association' or reboot the box.
IKE-DH Private Exponent	Ephemeral DH private exponent used in IKE. DH N = 224, 256 bit, ECDH P-256, or ECDH P-384	Plaintext	Memory	'clear security IKE security-association' command or reboot the box.
IKE-PSK	Pre-shared authentication key used in IKE.	Hashed	Virtual disk (mapped to SDD)/Memory	'clear security IKE security-association' command or reboot the box. Key values stored in flash are not zeroized unless an explicit "request system zeroize" is executed.
ecdh private keys	Loaded into memory to complete key exchange in session establishment	Plaintext	Memory	Memory free() operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)

Table 9 CSP Storage and Zeroization

68. Junos OS does not provide a CLI interface to permit the viewing of keys. Cryptographic keys are protected through the enforcement of kernel-level file access rights, limiting access to the contents of cryptographic key containers to processes with cryptographic rights or shell users with root permission[48]. (***FPT_SKP_EXT.1***)

7.1.2 Random Bit Generation

69. Junos OS performs random bit generation in accordance with NIST Special Publication 800-90 using HMAC_DRBG, SHA-256. The RBG for the (virtualised) SRX SRX1500, SRX4100, SRX4200 and SRX4600 Series platforms is seeded by one hardware-based and two software-based noise sources of entropy: RANDOM_INTERRUPT, RANDOM_PURE_RDRAND (RDRAND) and RANDOM_NET.

- RANDOM_INTERRUPT (software-based): This source of entropy is provided by devices whose hardware interrupts are known to provide some amount of entropy, such as hard drive controllers. The timings are fed into kernel HMAC DRBG (Juniper kernel DRBG) along with a CPU cycle counter. Entropy claimed per entropy harvest event is 1 bit. On interrupt from network DMA, 152 bytes of memory buffer representing internal packet representation gets hashed into 4 bytes using Jenkins_hash + 32 bits cpu cycle count.

[48] Security Administrators do <u>not</u> have root permission in shell.

- RANDOM_PURE_RDRAND (hardware-based): This hardware source of entropy provides 8 bytes of RDRAND[49] along with a CPU cycle counter are fed into kernel HMAC DRBG (Juniper kernel DRBG). Entropy claimed per entropy harvest event is 32 bits. This originates from 8 bytes from intel RDRAND instruction and 32 bits cpu cycle count.
- RANDOM_NET (software-based): This source of entropy is associated with network activity. Timings (CPU counter values at the time of the event) together with internal representation of the network packets are used to construct extra entropy that is fed in to kernel HMAC DRBG. Entropy claimed per entropy harvest event is 2 bits. 16 bytes of internal representation of interrupt gets hashed into 4 bytes using Jenkins_hash + 32 bits cpu cycle count.

7.1.3 SSH

70. Junos OS supports and enforces Trusted Channels that protect the communications between the TOE and a remote audit server from unauthorized disclosure or modification. It also supports Trusted Paths between itself and remote administrators so that the contents of administrative sessions are protected against unauthorized disclosure or modification. (***FTP_ITC.1, FTP_TRP.1/Admin***)

71. Junos OS provides an SSH server to support Trusted Channels using SSHv2 protocol which ensures the confidentiality and integrity of communication with the remote audit server. Export of audit information to a secure, remote server is achieved by setting up an event trace monitor that sends event log messages by using NETCONF over SSH to the remote system event logging server. The remote audit server initiates the connection. The SSHv2 protocol ensures that the data transmitted over a SSH session cannot be disclosed or altered by using the encryption and integrity mechanisms of the protocol with the FIPS cryptographic module. (***FTP_ITC.1, FCS_SSHS_EXT.1***)

72. The Junos OS SSH Server also supports Trusted Paths using SSHv2 protocol which ensures the confidentiality and integrity of user sessions. The encrypted communication path between Junos OS SSH Server and a remote administrator is provided by the use of an SSH session. Remote administrators of Junos OS initiate communication to the Junos CLI through the SSH tunnel created by the SSH session. Assured identification of Junos OS is guaranteed by using public key based authentication for SSH. The SSHv2 protocol ensures that the data transmitted over a SSH session cannot be disclosed or altered by using the encryption and integrity mechanisms of the protocol with the FIPS cryptographic module. If desired, an additional layer of protection can be afforded to the trusted path by using IPSec to encapsulate the SSH connection. (***FTP_TRP.1/Admin, FCS_SSHS_EXT.1***)

73. The Junos OS SSH server is implemented in accordance with RFCs 4251, 4252, 4253, 4254, 5656 and 6668. Junos OS provides assured identification of the Junos OS appliance though public key authentication and supports password-based authentication by administrative users (Security Administrator) for SSH connections. The following table identifies conformance to the SSH related RFCs:

[49] Taken from the Intel RDRAND hardware entropy source of physical platform on which Junos VM is loaded. The hypervisor exposes the instruction set for the entropy seeding to be fed directly into the Junos OS VM.

RFC	Summary	TOE implementation of Security
RFC 4251	The Secure Shell (SSH) Protocol Architecture	**Host Keys:** The TOE uses an ECDSA Host Key for SSH v2, with a key size of 256 bits or greater, which is generated on initial setup of the TOE. It can be de-configured via the CLI and the key will be deleted and thus unavailable during connection establishment. This key is randomly generated to be unique to each TOE instance. The TOE presents the client with its public key and the client matches this key against its known_hosts list of keys. When a client connects to the TOE, the client will be able to determine if the same host key was used in previous connections, or if the key is different (per the SSHv2 protocol). Junos OS also supports RSA-based key establishment schemes with a key size of 2048 bits. **Policy Issues:** The TOE implements all mandatory algorithms and methods. The TOE can be configured to accept public-key based authentication and/or password-based authentication. The TOE does not require multiple authentication mechanisms for users. The TOE allows port forwarding and sessions to clients. The TOE has no X11 libraries or applications and X11 forwarding is prohibited. **Confidentiality:** The TOE does not accept the “none” cipher. supports AES-CBC-128, AES-CBC-256, AES-CTR-128, AES-CTR-256 encryption algorithms for protection of data over SSH and uses keys generated in accordance with “ssh-rsa”, “ecdsa-sha2-nistp256”, “ecdsa-sha2-nistp384” or “ecdsa-sha2-nistp521” to perform public-key based device authentication. For ciphers whose blocksize >= 16, the TOE rekeys every (2^32-1) bytes. The client may explicitly request a rekeying event as a valid SSHv2message at any time and the TOE will honor this request. Re-keying of SSH session keys can be configured using the sshd_config knob. The data-limit must be between 51200 and 4294967295 (2^32-1) bytes and the time-limit must be between 1 and 1440 minutes. In the evaluated deployment the time-limit must be set within 1 and 60 minutes. **Denial of Service:** When the SSH connection is brought down, the TOE does not attempt to re-establish it. **Ordering of Key Exchange Methods:** Key exchange is performed only using one of the supported key exchange algorithms, which are ordered as follows: ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521 (all specified in RFC 5656), diffie-hellman-group14-sha1 (specified in RFC 4253). **Debug Messages:** The TOE sshd server does not support debug messages via the CLI. **End Point Security:** The TOE permits port forwarding. **Proxy Forwarding:** The TOE permits proxy forwarding. **X11 Forwarding:** The TOE does not support X11 forwarding.

RFC	Summary	TOE implementation of Security
RFC 4252	The Secure Shell (SSH) Authentication Protocol	**Authentication Protocol:** The TOE does not accept the "none" authentication method. The TOE implements a timeout period of 30seconds for authentication of the SSHv2 protocol and provides a limit of three failed authentication attempts before sending a disconnect to the client. **Authentication Requests:** The TOE does not accept authentication if the requested service does not exist. The TOE does not allow authentication requests for a non-existent username to succeed – it sends back a disconnect message as it would for failed authentications and hence does not allow enumeration of valid usernames. The TOE denies "none" authentication method and replies with a list of permitted authentication methods. **Public Key Authentication Method:** The TOE supports public key authentication for SSHv2 session authentication. The SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key. Authentication succeeds if the correct private key is used. The TOE does not require multiple authentications (public key and password) for users. **Password Authentication Method:** The TOE supports password authentication. Expired passwords are not supported and cannot be used for authentication. **Host-Based Authentication:** The TOE does not support the configuration of host-based authentication methods.
RFC 4253	The Secure Shell (SSH) Transport Layer Protocol	**Encryption:** The TOE offers the following for encryption of SSH sessions: aes128-cbc and aes256-cbc, aes128-ctr, aes256-ctr. The TOE permits negotiation of encryption algorithms in each direction. The TOE does not allow the "none" algorithm for encryption. **Maximum Packet length:** Packets greater than 256Kbytes in an SSH transport connection are dropped and the connection is terminated by Junos OS. **Data Integrity:** The TOE permits negotiation of HMAC-SHA1 in each direction for SSH transport. **Key Exchange:** The TOE supports diffie-hellman-group14-sha1. **Key Re-Exchange:** The TOE performs a re-exchange when SSH_MSG_KEXINIT is received.

RFC	Summary	TOE implementation of Security
RFC 4254	Secure Shell (SSH) Connection Protocol	**Multiple channels:** The TOE assigns each channel a number (as detailed in RFC 4251, see above). **Data transfers:** The TOE supports a maximum window size of 256K bytes for data transfer. **Interactive sessions:** The TOE only supports interactive sessions that do NOT involve X11 forwarding. **Forwarded X11 connections:** This is not supported in the TOE. **Environment variable passing:** The TOE only sets variables once the server process has dropped privileges. **Starting shells/commands:** The TOE supports starting one of shell, application program or command (only one request per channel). These will be run in the context of a channel, and will not halt the execution of the protocol stack. **Window dimension change notices:** The TOE will accept notifications of changes to the terminal size (dimensions) from the client. **Port forwarding:** This is fully supported by the TOE.
RFC5656	SSH ECC Algorithm Integration	**ECDH Key Exchange**: The support key exchange methods specified in this RFC are ecdh-sha2-nistp256, ecdh-sha2-nistp384, or ecdh-sha2-nistp521. The client matches the key against its known_hosts list of keys. **Hashing**: Junos OS supports cryptographic hashing via the SHA-256 and SHA-512 algorithms, provided it has a message digest size of either 256 or 512 bits.**Required Curves:** All required curves are implemented: ecdh-sha2-nistp256, ecdh-sha2-nistp384, or ecdh-sha2-nistp521. None of the Recommended Curves are supported as they are not included in [NDcPP].
RFC 6668	sha2-Transport Layer Protocol	**Data Integrity Algorithms:** Both the recommended and optional algorithms hmac-sha2-256 and hmac-sha2-512 (respectively) are implemented for SSH transport.

Table 10 SSH RFC conformance

74. Certificates are stored in non-volatile flash memory. Access to flash memory requires administrator credentials. A certificate may be loaded via command line (FIA_X.509_EXT.1).

7.1.4 IPsec

75. The TOE is conformant to RFC 4301. (***FCS_IPSEC_EXT.1.1***)

76. The TOE supports tunnel mode only. (***FCS_IPSEC_EXT.1.3***)

77. By default, the TOE denies all traffic through an SRX Series device. In fact, an implicit default security policy exists that denies all packets. You can change this behavior by configuring a standard security policy that permits certain types of traffic. The implicit default policy can be changed to permit all traffic with the '`set security policies default-policy`' command; however, this is not recommended.

78. The security policy rule set is an ordered list of security policy entries enforced by the firewall rules, each of which contains the specification of a network flow and an action:

- Source IP address and network mask
- Destination IP address and network mask
- Protocol

- Source port
- Destination port
- Action: permit, deny, drop silently, log

79. Each packet is compared against entries in the security policy ruleset in sequential order until one is found that matches the specification in the policy, or until the end of the rule set is reached, in which case the implicit default policy is implemented and the packet is discarded. (***FCS_IPSEC_EXT.1.2, supported by FPF_RUL_EXT.1.1, FPF_RUL_EXT.1.3, FPF_RUL_EXT.1.4, FPF_RUL_EXT.1.6, FPF_RUL_EXT.1.7***)

80. The TOE supports AES-GCM-128, AES-GCM-192 and AES-GCM-256, and AES-CBC-128, AES-CBC-192 or AES-CBC-256 using HMAC SHA-256 for ESP protection. (***FCS_IPSEC_EXT.1.4***)

81. IKEv1 and IKEv2 are implemented. IKEv1 as defined in RFCs 2407, 2408, 2409, RFC 4109 and RFC 4868 for hash functions; IKEv2 as defined in RFCs 5996 (with no support for NAT traversal) and RFC 4868 for hash functions. IKEv1 aggressive mode is not supported. (***FCS_IPSEC_EXT.1.5***)

82. The TOE supports AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128 and AES-GCM-256 for payload protection in IKEv1 and IKEv2. (***FCS_IPSEC_EXT.1.6***)

83. In the evaluated configuration, the TOE permits configuration of the IKE and IPsec lifetime exchanges in terms of number of (kilo)bytes (64 to 4294967294 kilo bytes) or length of time (180 to 86400 seconds):
(***FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8***)

84. The following CLI commands configure a lifetime of either kilobytes or seconds:
(***FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8***)

```
set security ipsec proposal <name> lifetime-kilobytes <kb>
set security ipsec proposal <name> lifetime-seconds <seconds>
```

85. The TOE supports Diffie-Hellman Groups 14, 19, 20 and 24. In the IKEv1 phase 1 and phase 2 exchanges, the TOE and peer will agree on the best DH group both can support. When the TOE receives an IKE proposal, it will select the first DH group that matches the acceptable DH groups configured in the TOE (one or more of DH Groups 14, 19, 20 or 24) and the negotiation will fail if there is no match. Similarly, when the peer initiates the IKE protocol, the TOE will select the first match from the IKE proposal sent by the peer and the negotiation fails is no acceptable match is found. (***FCS_IPSEC_EXT.1.11***)

86. 88. The TOE uses HMAC DRBG with SHA-256 for the generation of DH exponents and nonces in the IKE key exchange protocol of length 224 bits (for DH Group 14), 256 bits (for DH Groups 19 and 24) and 384 bits (for DH Group 20). (***FCS_IPSEC_EXT.1.9, FCS_IPSEC_EXT.1.10***)

87. The TOE supports both RSA and ECDSA for use with X.509v3 certificates that conform to RFC 4945 and pre-shared Keys for IPsec support. (***FCS_IPSEC_EXT.1.13***)

88. The TOE ensures that the strength of the symmetric algorithm (128, 192 or 256 bits) negotiated to protect the IKEv1 Phase 1, IKEv2 IKE_SA connection is greater than or equal to the strength of the symmetric algorithm negotiated to protect the IKEv1 Phase 2, IKEv2 CHILD_SA connection. (***FCS_IPSEC_EXT.1.12***)

89. The TOE uses pre-shared keys for IPSec. The TOE accepts Unicode characters to specify text-based pre-shared keys. Unicode characters are encoded as UTF-8 and treated as multiple bytes – up to 4 bytes depending on the character. The maximum length limit for text-based pre-shared keys enforced by the TOE is 255 bytes. When a pre-shared key is only composed of ASCII characters this limit is equivalent to 255 characters. The text-based pre-shared or bit-based keys may contain upper and lower case letters, numbers, and special characters (that include: “!”,

"@", "#", "$", "%", "^", "&", "*", "(", and ")". The TOE accepts pre-shared text keys and converts the text string into an authentication value as per RFC 2409 for IKEv1 or RFC 4306 for IKEv2, using the PRF that is configured as the hash algorithm for the IKE exchanges. (FIA_PSK_EXT.1).

90. The TOE uses X.509 certificates as defined in RFC 5280.

91. To generate a Certificate Request, the administrator uses the CLI command

```
request security pki generate-certificate-request
```

92. and supplies the following values:

- Certificate-id – The internal identifier string for this certificate
- Domain-name
- Email address
- IP address
- Subject (DC=<Domain component>,CN=<Common-Name>,OU=<Organizational-Unit-name>,O=<Organization-name>,SN=<Serial-Number>,L=<Locality>,ST=<state>,C=<Country>)
- Filename – The local file in which to store the certificate signing request

(***FIA_X509_EXT.3***)

93. To validate certificates, the TOE extracts the subject, issuer, subjects public key, signature, basicConstraints and validity period fields. If any of those fields is not present, the validation fails. The issuer is looked up in the PKI database. If the issuer is not present, or if the issuer certificate does not have the CA:true flag in the basicConstraints section, the validation fails. The TOE verifies the validity of the signature. If the signature is not valid, the validation fails. It then confirms that the current date and time is within the valid time period specified in the certificate. The TOE also extracts the extendedKeyUsage field and verifies the value represents that for the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3).

94. If the TOE is configured to perform a revocation check using CRL (as specified in RFC 5280 Section 6.3) and the CRL fails to download, there are two possible outcomes: If the TOE is configured with the option to skip CRL checking on download failure enabled, then the certificate shall be considered as having passed the validation. If the TOE is configured with the option to skip CRL checking on download failure disabled, then the certificate is considered to have failed validation.

95. The TOE validates a certificate path by building a chain of (at least 3) certificates based upon issuer and subject linkage, validating each according the certificate validation procedure described above. If any certificate in the chain fails validation, the validation fails as a whole. A self-signed certificate is not required to be at the root of the certificate chain.

96. The TOE determines if a certificate is a CA certificate by requiring the CA:true flag to be present in the basicConstraints section. (***FIA_X509_EXT.1.1/Rev***)

97. The TOE requires that the configured IKE identity of the local and remote endpoints to match the contents of the certificate associated with a SA endpoint. The TOE permits the identity to be expressed as distinguished name, fully qualified domain name (FQDN), user FQDN or IP address. If either certificate does not validate, or the contents do not match the configured identity, then the SA will not be established.

98. The PKI daemon on an SRX Series device validates all X509 certificates received from VPN peers during the IKE negotiation. If the TSF cannot establish a connection to determine the validity of a

certificate, the SA will not be established unless the administrator of the TOE has explicitly configured the TOE to disable the CRL check in case the connection can not be established.

99. (***FIA_X509_EXT.1/Rev, FIA_X509_EXT.2, FIA_PSK_EXT.1, FIA_X509_EXT.3***)

100. For public key-based authentication of IPsec connections, Junos OS validates the X.509 certificates by extracting the subject, issuer, signature, basicConstraints and validity period fields. If any of those fields is not present, the validation fails. The issuer is looked up in the PKI database. If the issuer CA is not present, or if the issuer certificate does not have the CA:true flag in the basicConstraints section, the validation fails. Junos OS verifies the validity of the signature. If the signature is not valid, the validation fails. It then confirms that the current date and time is within the valid time period specified in the certificate. (***FIA_X509_EXT.1/Rev, FIA_X509_EXT.2***

101. Junos OS generates Certificate Request Messages as specified in RFC 2986 when validating certificates for IPsec connections. Device-specific information, Common Name, Organization, Organizational Unit, Country and public key details are provided in the CSR. Junos OS validates the chain of certificates from the Root CA when the CA Certificate Response is received. (***FIA_X509_EXT.3***).

7.2 Administrator Authentication

102. Junos OS enforces binding between human users and subjects. The Security Administrator[50] is responsible for provisioning user accounts, and only the Security Administrator can do so. (***FMT_SMR.2***)

103. Junos users are configured under "system login user" and are exported to the password database '/var/etc/master.passwd'. A Junos user is therefore an entry in the password database. Each entry in the password database has fields corresponding to the attributes of "system login user", including username, (obfuscated) password and login class.

104. The internal architecture supporting Authentication includes an active process, associated linked libraries and supporting configuration data. The Authentication process and library are

- `login`()
- PAM Library module

105. Following TOE initialization, the `login`() process is listening for a connection at the local console. This 'login' process can be accessed through either direct connection to the local console or following successful establishment of a remote management connection over SSH, when a login prompt is displayed.

106. This login process identifies and authenticates the user using PAM operations. The login process does two things; it first establishes that the requesting user is whom they claim to be and second provides them with an interactive Junos Command interactive command line interface (CLI).

107. The SSH daemon supports public key authentication by looking up a public key in an authorized keys file located in the directory '.ssh' in the user's home directory (i.e. '~/.ssh/') and this authentication method will be attempted before any other if the client has a key available (***FIA_UIA_EXT.1***). The SSH daemon will ignore the authorized keys file if it or the directory '.ssh' or the user's home directory are not owned by the user or are writeable by anyone else.

108. For password authentication, login() interacts with a user to request a username and password to establish and verify the user's identity. The username entered by the administrator at the username prompt is reflected to the screen, but no feedback to screen is provided while the entry made by the administrator at the password prompt until the Enter key is pressed (***FIA_UAU.7***). `login`() uses PAM Library calls for the actual verification of this data. The

[50] The Security Administrator role is detailed in Section 7.6 below.

password is hashed and compared to the stored value, and success/failure is indicated to `login()`, (***FIA_UIA_EXT.1***). PAM is used in the TOE support authentication management, account management, session management and password management. Login primarily uses the session management and password management functionality offered by PAM.

109. The retry-options can be configured to specify the action to be taken if the administrator fails to enter valid username/password credentials for password authentication when attempting to authenticate via remote access. The retry-options are applied following the first failed login attempt for a given username (***FIA_AFL.1***). The length of delay (5-10 seconds) after each failed attempt is specified by the backoff-factor, and the increase of the delay for each subsequent failed attempt is specified by the backoff-threshold (1-3). The tries-before-disconnect sets the maximum number of times (1-10) the administrator is allowed to enter a password to attempt to log in to the device through SSH before the connection is disconnected. The lockout-period sets the amount of time in minutes before the administrator can attempt to log in to the device after being locked out due to the number of failed login attempts (1-43,200 minutes). It is also possible for another administrator to "unlock" the account of administrator whose account has been locked for a period of time following failed authentication attempts. Even when an account is blocked for remote access to the TOE, an administrator is always able to login locally through the serial console and the administrator can attempt authentication via remote access after the maximum timeout period of 24 hours.

110. The TOE requires users to provide unique identification and authentication data (passwords/public key) before any access to the system is granted. Prior to authentication, the only Junos OS managed responses provided to the administrator are (***FIA_UAU_EXT.2***):

- Negotiation of SSH session
- Display of the access banner
- ICMP echo responses.

111. Authentication data for fixed password authentication is a case-sensitive, alphanumeric value. The password has a minimum length of 10 characters and maximum length of 20 characters, and must contain characters from at least two different character sets (upper, lower, numeric, punctuation), and can be up to 20 ASCII characters in length (control characters are not recommended). Any standard ASCII, extended ASCII and Unicode characters can be selected when choosing a password. (***FIA_PMG_EXT.1***)

112. Locally stored authentication credentials are protected (***FPT_APW_EXT.1***):

- The passwords are stored in obfuscated form using HMAC-sha1.
- Authentication data for public key-based authentication methods are stored in a directory owned by the user (and typically with the same name as the user). This directory contains the files '.ssh/authorized_keys' and '.ssh/authorized_keys2' which are used for SSH public key authentication.

113. Junos enables Security Administrators to configure an access banner provided with the authentication prompt. The banner can provide warnings against unauthorized access to the TOE as well as any other information that the Security Administrator wishes to communicate. (***FTA_TAB.1***)

114. User sessions (local and remote) can be terminated by users (***FTA_SSL.4***). The administrative user can logout of existing session by typing logout to exit the CLI admin session and the Junos OS makes the current contents unreadable after the admin initiates the termination. No user activity can take place until the user re-identifies and authenticates.

115. The Security Administrator can set the TOE so that a user session is terminated after a period of inactivity. (***FTA_SSL_EXT.1, FTA_SSL.3***) For each user session Junos OS maintains a count of clock cycles (provided by the system clock) since last activity. The count is reset each time there is activity related to the user session. When the counter reaches the number of clock cycles equating to the configured period of inactivity the user session is locked out.

116. Junos OS overwrites the display device and makes the current contents unreadable after the local interactive session is terminated due to inactivity, thus disabling any further interaction with the TOE. This mechanism is the inactivity timer for administrative sessions. The Security Administrator can configure this inactivity timer on administrative sessions after which the session will be logged out.

7.3 Correct Operation

117. The Junos OS VM will run the following set of self-tests during power on to check the correct operation of the Junos OS VM portions of the TOE (***FPT_TST_EXT.1***):

- <u>Power on test</u> – determines the boot-device responds, and performs a memory size check to confirm the amount of available memory.
- <u>File integrity test</u> –verifies integrity of all mounted signed packages, to assert that system files have not been tampered with. To test the integrity of the firmware, the fingerprints of the executables and other immutable files are regenerated and validated against the SHA1 fingerprints contains in the manifest file.
- <u>Crypto integrity test</u> – checks integrity of major CSPs, such as SSH hostkeys and iked credentials, such as Cas, CERTS, and various keys.
- <u>Authentication error</u> – verifies that veriexec is enabled and operates as expected using /opt/sbin/kats/cannot-exec.real.
- <u>Kernel, libmd, OpenSSL, QuickSec, SSH Ipsec</u> – verifies correct output from known answer tests for appropriate algorithms.

118. Juniper Networks devices run only binaries supplied by Juniper Networks. Within the package, each Junos OS firmware image includes fingerprints of the executables and other immutable files. Junos firmware will not execute any binary without a validating a fingerprint. This feature protects the system against unauthorized firmware and activity that might compromise the integrity of the device. These self-tests ensure that only authorized executables are allowed to run thus ensuring the correct operation of the TOE.

119. In the event of a transiently corrupt state or failure condition, the system will panic; the event will be logged and the system restarted, having ceased to process network traffic. When the system restarts, the system boot process does not succeed without passing all applicable self-tests. This automatic recovery and self-test behavior, is discussed in Chapter 11 of the [ECG].

120. When any self-test fails, the device halts in an error state. No command line input or traffic to any interface is processed. The device must be power cycled to attempt to return to operation. This self-test behavior, is discussed in [ECG]. (***FPT_FLS.1, FPT_TST_EXT.1, FPT_TST_EXT.3***)

7.4 Trusted Update

121. Security Administrators are able to query the current version of the TOE firmware using the CLI command “show version” (***FPT_TUD_EXT.1***) and, if a new version of the TOE firmware is available, initiate an update of the TOE firmware. Junos OS does not provide partial updates for the TOE, customers requiring updates must migrate to a subsequent release. Updates are downloaded and applied manually (there is no automatic updating of the Junos OS). (***FPT_TUD_EXT.1, FMT_SMF.1, FMT_MOF.1/ManualUpdate,***)

122. The installable firmware package includes both the Junos OS VM and the WRL virtualization kernel. These cannot be updated separately in the evaluated configuration; they must be installed as a single package. Once the Junos OS VM is loaded the procedures detailed in [ECG] will be applied to disable the loading of additional VMs.

123. The installable software packages have a digital signature that is checked when the Security Administrator attempts to install the package. The signature of the complete package is verified at the beginning of the installation process before the package is expanded. If signature verification fails, an error message is displayed and the package is not installed.

124. The Junos OS kernel maintains a set of fingerprints (SHA1 digests) for executable files and other files which should be immutable, as described in Section 7.3. The manifest file is signed using the Juniper package signing key, and is verified by the TOE using the public key (stored on the TOE filesystem in clear, protected by filesystem access rights). ECDSA (P-256) with SHA-256 is used for digital signature package verification.

125. The fingerprint loader will only process a manifest for which it can verify the signature. Thus without a valid digital signature an executable cannot be run. When the command is issued to install an update, the manifest file for the update is verified and stored, and each executable/immutable file is verified before it is executed. If any of the fingerprints in an update are not correctly verified, the TOE uses the last known verified image.

126. A certificate may be loaded via command line and is stored in SSD. Access to flash memory requires administrator credentials. Control on access to the trust store holding the X509v3 certificates can be controlled using standard Junos permissions settings. Each top-level CLI command and each configuration statement have an access privilege level associated with them and users can execute only those commands and configure and view only those statements for which they have access privileges. The access privileges for each login class are defined by one or more permission flags. For each login class, the use of operational and configuration mode commands that would otherwise be permitted or not allowed by a privilege level specified in the permissions statement can be explicitly denied or allowed. Cryptographic keys are protected through the enforcement of kernel-level file access rights, limiting access to the contents of cryptographic key containers to processes with cryptographic rights. The TOE does not provide a CLI interface to permit the viewing of keys. (**FIA_X.509_EXT.1/Rev, FMT_MTD.1/CoreData**).

127. Junos OS verifies the validity of the signature. If the signature is not valid, the validation fails. If the signature is valid the update process proceeds. (**FCS_COP.1/SigGen, FPT_TUD_EXT.1**)

7.5 Audit

128. Junos OS creates and stores audit records for the following events (the detail of content recorded for each audit event is detailed in Table 4 (***FAU_GEN.1/ND***) and Table 5 (***FAU_GEN.1/IPS***). Auditing is implemented using syslog.

- Start-up and shut-down of the audit functions
- Administrative login and logout
- Configuration is committed
- Configuration is changed (includes all management activities of TSF data)
- Generating/import of, changing, or deleting of cryptographic keys (see below for more detail)
- Resetting passwords
- Starting and stopping services

- All use of the identification and authentication mechanisms
- Unsuccessful login attempts limit is met or exceeded
- Any attempt to initiate a manual update
- Result of the update attempt (success or failure)
- The termination of a local/remote/interactive session by the session locking mechanism
- Initiation/termination/failure of the SSH trusted channel to syslog server
- Initiation/termination/failure of the SSH trusted path with Admin
- Initiation/termination/failure of an IPsec trusted channel, including Session Establishment with peer
- Session establishment with CA
- Application of firewall rules configured with the ‘log’ operation by the stateful traffic filtering function
- Indication of packets dropped due to too much network traffic by the stateful traffic filtering function
- Application of rules configured with the ‘log’ operation by the packet filtering function
- Indication of packets dropped due to too much network traffic by the packet filtering function
- Start-up and shut-down of the IPS functions
- All dissimilar IPS events and reactions
- Totals of similar events and reactions occurring within a specified time period
- Modification of an IPS policy element
- Modification of which IPS policies are active on a TOE interface
- Enabling/disabling a TOE interface with IPS policies applied
- Modification of which mode(s) is/are active on a TOE interface
- Inspected traffic matches a list of known-good or known-bad addresses applied to an IPS policy
- Inspected traffic matches a signature-based IPS policy with logging enabled
- Inspected traffic matches an anomaly-based IPS policy

129. In addition the following management activities of TSF data are recorded:

- configure the access banner;
- configure the session inactivity time before session termination;
- configure the authentication failure parameters for FIA_AFL.1;
- Ability to configure audit behaviour;
- configure the cryptographic functionality;
- configure thresholds for SSH rekeying;
- re-enable an Administrator account;

- set the time which is used for time-stamps.

130. The detail of what events are to be recorded by syslog are determined by the logging level specified the "level" argument of the "`set system syslog`" CLI command. To ensure compliance with the requirements the audit knobs detailed in [ECG] must be configured.

131. As a minimum, Junos OS records the following with each log entry:

    - date and time of the event and/or reaction
    - type of event and/or reaction
    - subject identity (where applicable)
    - the outcome (success or failure) of the event (where applicable).

132. Because of the nature of IPS event logs, log generation often happens in bursts and can generate a large volume of messages during an attack. To manage the volume of log messages, Junos supports log suppression, which suppresses multiple instances of the same log occurring from the same or similar sessions over the same period of time. IPS log suppression is enabled by default and can be customized based on the following configurable attributes:

    - Source/destination addresses;
    - Number of log occurrences after which log suppression begins;
    - Maximum number of logs that log suppression can operate on;
    - Time after which suppressed logs are reported.

133. Suppressed logs are reported as single log entries containing the count of occurrences.

134. In order to identify the key being operated on, the following details are recorded for all administrative actions relating to cryptographic keys (generating, importing, changing and deleting keys):

    - PKID – certificate id will be recorded when generating or deleting a key pair
    - IKE SPI – IP address of the initiator and responder recorded, together with the SPI, will be recorded when generating a key pair. The IP address of the initiator and responder will provide the unique link to the key identifier (SPI) of the key that has been destroyed in the session termination
    - SSH session keys– key reference provided by process id
    - SSH keys ***generated*** for outbound trusted channel to external syslog server
    - SSH keys ***imported*** for outbound trusted channel to external syslog server
    - SSH key configured for SSH public key authentication –the hash of the public key that is to be used for authentication is recorded in syslog

135. For SSH (ephemeral) session keys the PID is used as the key reference to relate the key generation and key destruction audit events. The key destruction event is recorded as a session disconnect event. For example, key generation and key destruction events for a single SSH session key would be reflected by records similar to the following:

    Sep 27 15:09:36 yeti sshd[6529]: Accepted publickey for root from 10.163.18.165 port 45336 ssh2: RSA SHA256:l1vri77TPQ4VaupE2NMYiUXPnGkqBWIgD5vW0OuglGI
    …
    Sep 27 15:09:40 yeti sshd[6529]: Received disconnect from 10.163.18.165 port 45336:11: disconnected by user

Sep 27 15:09:40 yeti sshd[6529]: Disconnected from 10.163.18.165 port 45336

136. SSH keys ***generated*** for outbound trusted channels are uniquely identified in the audit record by the public key filename and fingerprint. For example:

Sep 27 23:36:49 yeti ssh-keygen [67873]: Generated SSH key file /root/.ssh/id_rsa.pub with fingerprint SHA256:g+7lsR7x4lQb1JT8Q3scfb2sOl8lyccojGdmkmw4dwM

137. SSH keys ***imported*** for use in establishing outbound trusted channels are uniquely identified in the audit record by the hash of the key imported and the username importing (to which the key will be bound).

138. It should be noted that SSH keys used for trusted channels are NOT deleted by mgd when SSH is de-configured. Hence, the only time SSH keys used for trusted channels are deleted is when a "request system zeroize hypervisor" action is performed and the whole VM is zeroized (which by definition cannot be recorded)

139. All events recorded by syslog are timestamped. The clock function of Junos OS provides a source of date and time information for the appliance, used in audit timestamps. Wind River Linux kernel provides the current time when it bootstraps the Junos OS VM. Once the Junos OS VM is started it maintains its own time using the hardware Time Stamp Counter as the clock source[51]. (***FAU_GEN.2, FPT_STM.1***)

140. Syslog can be configured to store the audit logs locally (***FAU_STG_EXT.1***), and optionally to send them to one or more syslog log servers via Netconf over SSH (***FAU_STG.1, FMT_MOF.1/Functions***). Local audit log are stored in /var/log/ in the underlying filesystem. Only a Security Administrator can read log files, or delete log and archive files through the CLI interface or through direct access to the filesystem having first authenticated as a Security Administrator. The syslogs are automatically deleted locally according to configurable limits on storage volume. The default maximum size is 1Gb. The default maximum size can be modified by the user, using the "`size`" argument for the "`set system syslog`" CLI command.

141. The Junos OS defines an active log file and a number of "archive" files (10 by default, but configurable from 1 to 1000). When the active log file reaches its maximum size, the logging utility closes the file, compresses it, and names the compressed archive file 'logfile.0.gz'. The logging utility then opens and writes to a new active log file. When the new active log file reaches the configured maximum size, 'logfile.0.gz' is renamed 'logfile.1.gz', and the active log file is closed, compressed, and renamed 'logfile.0.gz'. When the maximum number of archive files is reached and when the size of the active file reaches the configured maximum size, the contents of the oldest archived file are deleted so the current active file can be archived.

142. A 1Gb syslog file takes approximately 0.25Gb of storage when archived. Syslog files can acquire complete storage allocated to /var filesystem, which is platform specific. However, when the filesystem reaches 92% storage capacity an event is raised to the administrator but the eventd process (being a privileged process) still can continue using the reserved storage blocks. This allows the syslog to continue storing events while the administrator frees the storage. If the administrator does not free the storage in time and the /var filesystem storage becomes exhausted a final entry is recorded in the log reporting "No space left on device" and logging is terminated. The appliance continues to operate in the event of exhaustion of audit log storage space.

[51] Junos VM uses a tick count to maintain the "wall clock" within the VM, which reflects the "apparent" time (current time) from that passes in by the host when the VM is powered on.

7.6 Management

143. Accounts assigned to the Security Administrator role are used to manage Junos OS in accordance with [NDcPP]. User accounts in the TOE have the following attributes: user identity (user name), authentication data (password) and role (privilege). The Security Administrator is associated with the defined login class "security-admin", which has the necessary permission set to permit the administrator to perform all tasks necessary to manage Junos OS in accordance with the requirements of [NDcPP].(***FMT_SMR.2***)

144. The TOE provides user access either through the system console or remotely over the Trusted Path using the SSHv2 protocol. Users are required to provide unique identification and authentication data before any access to the system is granted, as detailed in Section 7.2 above. (***FMT_SMR.2, FMT_SMF.1***)

145. The Security Administrator has the capability to:

- Administer the TOE locally via the serial ports on the physical device or remotely over an SSH connection.
- Initiate a manual update of TOE firmware (***FMT_MOF.1/ManualUpdate***):
  - Query currently executing version of TOE firmware (***FPT_TUD_EXT.1***)
  - Verify update using digital signature (***FPT_TUD_EXT.1***)
- Manage Functions:
  - <u>Transmission of audit data to an external IT entity,</u> including Start/stop and modify the behaviour of the trusted communication channel to external syslog server (netconf over SSH) and the trusted path for remote Administrative sessions (SSH) (***FMT_MOF.1/Functions, FMT_MOF.1/Services, FMT_SMF.1***)
  - <u>Handling of audit data</u>, including setting limits of log file size (***FMT_MOF.1/Functions***)
- Manage TSF data (***FMT_MTD,1/CoreData***)
  - Create, modify, delete administrator accounts, including configuration of authentication failure parameters
  - Reset administrator passwords
  - Re-enable an Administrator account (***FIA_AFL.1***);
- Manage crypto keys (***FMT_MTD.1/CryptoKeys***):
  - SSH key generation (ecdsa, ssh-rsa)
- Perform management functions (***FMT_SMF.1***):
  - Configure the access banner (***FTA_TAB.1***)
  - Configure the session inactivity time before session termination or locking, including termination of session when serial console cable is disconnected (***FTA_SSL_EXT.1, FTA_SSL.3***)
  - Ability to import X.509v3 certificates (***FCS_IPSEC_EXT.1***)
  - Manage cryptographic functionality (***FCS_SSHS_EXT.1***), including:
    - ssh ciphers
    - hostkey algorithm
    - key exchange algorithm

        - hashed message authentication code
        - thresholds for SSH rekeying
    - Set the system time (***FPT_STM_EXT.1***)
    - Ability to configure Firewall rules (***FFW_RUL_EXT.1***);
    - Ability to configure the VPN-associated cryptographic functionality (**FCS_COP.1/DataEncryption, FCS_CKM.1.1/IKE*, FCS_IPSEC_EXT.1***);
    - Ability to configure the IPsec functionality (***FCS_IPSEC_EXT.1***), including configuration of IKE lifetime-seconds (within range 180 to 86400[52], with default value of 180 seconds), IPsec lifetime-seconds (within range 180 to 86400, with default value of 28800 seconds[53]), and Lifetime-kilobytes (within range 64 to 4294967294 kilobytes) and ability to configure the reference identifier for the peer;
    - Ability to enable, disable, determine and modify behavior, and configure all other VPN-associated security functions of the TOE identified in [VPN_EP] (***FPF_RUL_EXT.1, FCS_COP.1/DataEncryption, FCS_CKM.1.1/IKE, FCS_IPSEC_EXT.1***);
    - Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality (***IPS_NTA_EXT.1, IPS_IPB_EXT.1, IPS_SBD_EXT.1, IPS_ABD_EXT.1***)
    - Modify these parameters that define the network traffic to be collected and analysed (***IPS_NTA_EXT.1***):
        - Source IP addresses (host address and network address);
        - Destination IP addresses (host address and network address);
        - Source port (TCP and UDP);
        - Destination port (TCP and UDP);
        - Protocol (IPv4 and IPv6)
        - ICMP type and code
    - Update (import) IPS signatures (***IPS_SBD_EXT.1***);
    - Create custom IPS signatures (***IPS_SBD_EXT.1***);
    - Configure anomaly detection (***IPS_ABD_EXT.1***);
    - Enable and disable actions to be taken when signature or anomaly matches are detected (***IPS_SBD_EXT.1***);
    - Modify thresholds that trigger IPS reactions (***IPS_ABD_EXT.1***);
    - Modify the duration of traffic blocking actions (***IPS_ABD_EXT.1***);
    - Modify the known-good and known-bad lists (of IP addresses or address ranges) (***IPS_IPB_EXT.1***);
    - Configure the known-good and known-bad lists to override signature-based IPS policies (***IPS_SBD_EXT.1***).

146. Detailed topics on the secure management of Junos OS are discussed in [ECG].

7.7 User Data Protection

147. The only resource made available to information flowing through a TOE is the temporary storage of packet information when access is requested and when information is being routed. User data is not persistent when resources are released by one user/process and allocated to another user/process. Temporary storage (memory) used to build network packets is erased when the resource is called into use by the next user/process. Junos knows, and keeps track of, the length of the packet. This means that when memory allocated from a previous user/process arrives to build the next network packet, Junos is aware of when the end of the packet is reached and pads

[52] 180 to 86400 seconds rang is a range of 3 minutes to 24 hours.
[53] 28800 seconds is 8 hours.

a short packet with zeros accordingly. Therefore, no residual information from packets in a previous information stream can traverse through the TOE. (***FDP_RIP.2***)

7.8 Packet Filtering/Stateful Traffic

148. The boot sequence of the TOE appliances also aids in establishing the securing domain and preventing tampering or bypass of security functionality. This includes ensuring the packet filtering rules cannot be bypassed during the boot sequence of the TOE. The following steps list the boot sequence for the TOE:

- BIOS hardware and memory checks
- Loading and initialization of the FreeBSD Kernel OS
- FIPS self-tests and firmware integrity tests are executed
- The init utility is started (mounts file systems, sets up network cards to communicate on the network, and generally starts all the processes that usually are run on a FreeBSD system at startup)
- Daemon programs such as Internet Service Daemon (INETD), Routing Protocol Daemon (RPD), Syslogd are started; Routing and forwarding tables are initialized
- Management Daemon (or MGD) is loaded, allowing access to management interface
- Physical interfaces are active

149. Once the interfaces are brought up, they will start to receive and send packets based on the current configuration (or not receive or send any packets if they have not been previously configured). Interfaces are brought up only after successful loading of kernel and Information Flow subsystems, and these interfaces cannot send or receive packets unless previously configured by an Administrator. Since the Management Daemon is not loaded until after the kernel and INETD are initialized, no modification to the security attributes can be made by a user or process other than via the management process.

150. The trusted and untrusted network connection interfaces on the security appliance are not enabled until all of the components on the appliance are fully initialized; power-up tests are successful and ready to enforce the configured security policies. In this manner, the TOE ensures that Administrators are appropriately authorized when they exercise management commands and any network traffic is always subject to the configured information flow policies.

151. The TOE is configured to associate network interfaces to IP subnets. Source IP addresses are then associated with the network interface.

152. Junos is composed of a number of separate executables, or daemons. If a failure occurs in the “flow” daemon (flowd) causing it to halt, no packet processing will occur and no packets will be forwarded. A failure in another daemon will not prevent the flow daemon from enforcing the policy rule set.

153. The Information Flow subsystem is responsible for processing the arriving packets from the network to the TOE's network interface. Based on Administrator-configured policy, interface and zone information, the packet flows through the various modules of the Information Flow subsystem. Rules within policies are processed in an Administrator-defined order when network traffic flows through the TOE network interfaces. By default, the TOE behavior is to deny packets when there is no rule match unless another required condition allows the network traffic If a security risk is found in the packet. e.g. denial-of-service attacks, the packet is dropped and an event is logged. The packet does not continue to the next module for processing. If the packet is not dropped by a given module, the interrupt handling routine calls the function for the next relevant module.

154. In case of an interface getting overwhelmed, packets are dropped. This is recorded by the SNMP mibs as well as a log. When an interface gets overwhelmed with CPU utilization 99% then packets are dropped with syslog record as '`CPU Utilization greater than 99, expect packet loss`'.

155. The Information Flow subsystem consists of the following modules:

- IP Classification Module
- Attack Detection Module
- Session Lookup Module
- Security Policy Module
- Session Setup Module
- Inetd Module
- Rdp Module

156. The IP Classification module retrieves information from packets received on the network interface device, classifies packets into several categories, saves classification information in packet processing context, and provides other modules with that information for assisting further processing.

157. The Attack Detection module provides inline attack detection such as IP Spoofing for the security appliance. This module monitors arriving traffic, performs predefined attack detection services (prevents attacks), and issues actions when an attack is found.

158. The Session Lookup module performs lookups in the session table which is used for all interfaces based on the information in incoming packets. Specifically, the lookup is based on the exact match of source IP address and port, destination IP address and port, protocol attributes (e.g., SYN, ACK, RST, and FIN), and egress/ingress zone. The input is passed to the module as a set of parameters from the Attack Detection module via a function call. The module returns matching wing if a match is found and 0 otherwise. Sessions are removed when terminated.

159. The Session Setup module is only available for packets that do not match current established sessions. It is activated after the Session Lookup module. If packet has a matched session, it will skip the session setup module and proceed to the Security Policy module, and other modules. Eventually if the packet is not destined for the TOE, the Network interface will pass the traffic out of the appliance.

160. The Security Policy module examines traffic passing through the TOE (via Session Setup module) and determines if the traffic can pass based on administrator-configured access policies. The Security Policy module is the core of the firewall and IPS functionalities in the TOE: It is the policy enforcement engine that fulfills the security requirements for the user. The Security Policy module will deny packets when there is no policy match unless another policy allows the traffic.

161. The Session Setup module performs the auditing of denied packets. If there is a policy to specifically deny traffic, traffic matching this deny policy is dropped and logged in traffic log. If there is no policy to deny traffic, traffic that does not match any policy is dropped and not logged. In either case, Session Setup module does not create any sessions for denied traffic. Sessions are created for allowed traffic.

162. The INETD module provides internet services for the TOE. The module listens on designated ports used by internet services such as FTP. When a TCP or UDP packet arrives with a particular destination port number, INETD launches the appropriate server program (e.g., SSHD) to handle the connection.

163. The RPD (Routing Protocol Daemon) module provides the implementations and algorithms for the routing protocols and route calculations. The primary goal of the RPD is to create and maintain the Routing Information Base (RIB), which is a database of routing entries. Each routing entry consists of a destination address and some form of next hop information. RPD module maintains the internal routing table and properly distributes routes from the routing table to Kernel subsystem used for traffic forwarding at the Network interface.

164. The TOE performs stateful network traffic filtering on network packets using the following network traffic protocols and network fields conforming to the described RFCs:

PROTOCOL/RFC	FIELDS
Internet Control Message Protocol version 4 (ICMPv4) RFC 792 (ICMPv4)	Type Code
Internet Control Message Protocol version 6 (ICMPv6) RFC 4443 (ICMPv6)	Type Code
Internet Protocol (IPv4) RFC 791 (IPv4)	Source address Destination Address Transport Layer Protocol
Internet Protocol version 6 (IPv6) RFC 2460 (IPv6)	Source address Destination Address Transport Layer Protocol
Transmission Control Protocol (TCP) RFC 793 (TCP)	Source port Destination port
User Datagram Protocol (UDP) RFC 768 (UDP)	Source port Destination port

Table 11 Traffic Filtering RFCs

165. Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.

166. The TOE shall allow permit, deny, and log operations to be associated with rules and these rules can be assigned to distinct network interfaces.

167. The TOE accepts network packets if it matches an established TCP, UDP or ICMP session using:

- TCP: source and destination addresses, source and destination ports, sequence number, flags
- UDP: source and destination addresses, source and destination ports
- ICMP: source and destination addresses, type, code

168. The TOE will remove existing traffic flows due to session inactivity timeout, or completion of the session.

169. The TOE supports FTP (RFC 959) to dynamically establish sessions allowing network traffic according to Administrator rules. Session events will be logged in accordance with 'log' operations defined in the rules. Source and destination addresses, source and destination ports, transport layer protocol, and TOE Interface are recorded in each log record. (**FFW_RUL_EXT.2**)

170. Junos implements what is referred to as an Application Layer gateway (ALG) that inspects FTP traffic to determine the port number used for data sessions. The ALG permits data traffic for the duration of the session, closing the port when the session ends. In this context, "session" refers to the TCP data transfer connection, not the duration of the FTP control session. Junos implements ALGs for a number of protocols.

171. The TOE enforces the following default reject rules with logging on all network traffic:

- invalid fragments;
- fragmented IP packets which cannot be re-assembled completely;
- where the source address is equal to the address of the network interface where the network packet was received;
- where the source address does not belong to the networks associated with the network interface where the network packet was received;
- where the source address is defined as being on a broadcast network;
- where the source address is defined as being on a multicast network;
- where the source address is defined as being a loopback address;
- where the source address is a multicast;
- packets where the source or destination address is a link-local address;
- where the source or destination address is defined as being an address “reserved for future use” as specified in RFC 5735 for IPv4;
- where the source or destination address is defined as an “unspecified address” or an address “reserved for future definition and use” as specified in RFC 3513 for IPv6;
- with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified;
- packets are checked for validity. “Invalid fragments” are those that violate these rules:
    - No overlap
    - The total fragments in one packet should not be more than 62 pieces
    - The total length of merged fragments should not larger than 64k
    - All fragments in one packet should arrive in 2 seconds
    - The total queued fragments has limitation, depending on the platform
    - The total number of concurrent fragment processing for different packet has limitations depending on platform

172. The TOE can be configured to drop connection attempts after a defined number of half-open TCP connections using the Junos screen ‘tcp syn-flood’, which provides both source and destination thresholds on the number of uncompleted TCP connections, as well as a timeout period. The source threshold option allows administrators to specify the number of SYN segments received per second from a single source IP address—regardless of the destination IP address—before Junos OS begins dropping connection requests from that source. Similarly, the destination threshold option allows administrators to specify the number of SYN segments received per second for a single destination IP address before Junos OS begins dropping connection requests to that destination. The timeout option allows administrators to set the maximum length of time before an uncompleted connection is dropped from the queue.

173. For more information about configuring event logging, see Junos OS Complete Software Guide for SRX Series Services Gateways (Volume 2), Guide 4: ‘Building Blocks Feature Guide for Security Devices’ and [ECG] (***FFW_RUL_EXT.1, FPF_RUL_EXT.1***)

7.9 Intrusion Prevention System

174. The Junos OS Intrusion Detection and Prevention (IDP) policy enables selectively enforcing various attack detection and prevention techniques on network traffic passing through an IDP-enabled device. Policy rules can be defined to match a section of traffic based on a zone, network, and application, and then take active or passive preventive actions on that traffic.

175. An IDP policy is made up of rule bases, and each rule base contains a set of rules that specify rule parameters, such as traffic match conditions, action, and logging requirements. IDP policies can then be associated to firewall policies. IDP can be invoked on a firewall rule by rule basis for maximum granularity. Only firewall policies marked for IDP will be processed by IDP engine, all other rules will only be processed by the firewall[54].

176. Firewall Policies match Source Zone, Destination Zone, Source IP, Destination IP, Source Port, Destination Port, and Protocol. Interface and VLAN matching can be achieved through the use of zones. Rules are organized into a firewall policy rulebase. Within IPS Policies, further matching for specific attacks is done on Source Zone, Destination Zone, Source IP, Destination IP, Source Port, Destination Port, and Protocol. Interface matching can be achieved through the use of zones. Attack Actions are configurable on a rule by rule basis. Rules within policies are processed in an Administrator-defined order when network traffic flows through the TOE network interfaces. (***IPS_NTA_EXT.1.1***)

177. Once stateful firewall processing of packets has been performed by the Information Flow subsystem, if a firewall policy that has been marked for IDP processing is triggered, the packets are processed by the IPS subsystem as follows:

- Fragmentation Processing – IP Fragments are reordered and reassembled. Duplicate, over/undersized, overlapping, incomplete and other invalid fragments are discarded.
- Flow Module SSL Decryption – sessions are checked for existing IP Actions, if none exist, new sessions are created. If a destination is marked for SSL decryption, a copy of the SSL traffic will be sent to the decryption engine. The original packet will be queue until inspection is complete.
- Packet Serialization and TCP Reassembly – packets are ordered and all TCP packets are reassembled into complete application messages.
- Application ID – pattern matching is performed on the traffic to determine what application the traffic is. The traffic is still inspected for Attacks, even if application cannot be determined.
- Protocol Decoding – protocol parsing and decoding is performed. Messages are deconstructed into application “contexts” which identify components of messages. Protocol Anomaly Detection is performed, along with AppDoS (if configured) by thresholds of these contexts.
- Attack Signature Matching – signatures are detected via deterministic finite automaton (DFA) pattern matching.
- IDP Attack Actions – when an attack is detected the corresponding policy configured action is executed. Possible actions include:
    - No Action
    - Drop packet

[54] Note that some of the security functionality required by the IPS EP is implemented at the firewall level without intervention of Junos IDP engine.

  - Drop connection
  - Close client (send an RST packet to the client)
  - Close server (sends an RST packet to the server)
  - Close client and server (sends an RST packet to both client and server)

178. The TOE supports stateful signature based attack detection defined as Attack Objects. Attack Objects use context based matching to match regular expressions in specific locations where they occur. Attack Objects can be composed of multiple signatures and protocol anomalies, including logical expressions between signatures for compound matching.

179. As indicated in Section 7.8 the TOE is capable of inspecting IPv4, IPv6, ICMPv4, TCP and UPD traffic. Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team. (***IPS_NTA_EXT.1.2***)

180. The TOE is capable of inspecting all traffic passing through the TOE's Ethernet interfaces (inline mode). Ethernet interfaces can be assigned to Zones on which firewall and IDP policies are predicated. The TOE supports management through the console port, as well as through a dedicated Ethernet management port whose traffic is never processed for routing. Remote management of the TOE can also be performed via SSH as described in Section 7.1.3. (***IPS_NTA_EXT.1.3***)

181. The TOE supports the definition of known-good and known-bad lists of source and/or destination addresses at the firewall rule level as described in Section 7.8. Address ranges can be defined by creating address book entries and attaching them to firewall policies. (***IPS_IPB_EXT.1***)

182. IPS signatures (in the sense of the IPS EP) are articulated at different points along the traffic processing flow implemented in the TOE. In Junos OS, interfaces are grouped into zones. The TOE supports the definition of signatures at the zone level, also known as the screen level. Junos OS screen options secure a zone by inspecting, then allowing or denying, all connection attempts that require crossing an interface bound to that zone. Sanity checks on IPv4 and IPv6 aimed at detecting malformed packets are performed at the screen level. In addition to attack detection and prevention at the screen level, Junos OS implements firewall and IDP policies at the inter-, intra-, and super-zone policy levels (super-zone here means in global policies, where no security zones are referenced). The TOE supports inspection of the following packet header information:

- IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; and IP Options.
- IPv6: Version; traffic class; flow label; payload length; next header; hop limit; source address; destination address; routing header; home address options.
- ICMP: Type; Code; Header Checksum; and Rest of Header (varies based on the ICMP type and code).
- ICMPv6: Type; Code; and Header Checksum.
- TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.
- UDP: Source port; destination port; length; and UDP checksum.

183. Signatures can be defined to match the any of above header-field values, using the command "set security idp custom-attack", along with the actions (allow/block), using the command "set security idp idp-policy", that the TOE will perform when a match is found in the processed packets. The matching criteria can be "equal", "greater-than", "less-than" or "not-equal". (***IPS_SBD_EXT.1.1***)

184. The TOE also supports string-based pattern-matching inspection of packet payload data for the above listed protocols. For TCP payload inspection, Junos OS provides pre-defined attack signatures to detect FTP commands, HTTP commands and content, and STMP states. Alternative, administrators can define custom-attack signatures for these application layer protocols using the command "set security idp custom-attack". (***IPS_SBD_EXT.1.2***)

185. The TOE is capable of detecting the following signatures using Junos predefined screen options:

IPS EP signature name	Junos screen name
IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)	ip tear-drop
IP source address equal to the IP destination (Land attack)	tcp land
Fragmented ICMP Traffic (e.g. Nuke attack)	icmp fragment
Large ICMP Traffic (Ping of Death attack)	icmp ping-death
TCP NULL flags	tcp tcp-no-flag
TCP SYN+FIN flags	tcp syn-fin
TCP FIN only flags	tcp fin-no-ack
UDP Bomb Attack	udp length-error
ICMP flooding (Smurf attack, and ping flood)	icmp flood
TCP flooding (e.g. SYN flood)	tcp syn-flood
IP protocol scanning	ip unknown-protocol
TCP port scanning	tcp port-scan
UDP port scanning	udp port-scan
ICMP scanning	icmp ip-sweep

Table 12 IPS signature names

186. The default action for the above screens is to drop the packets. To allow the packets through, the "alarm-without-drop" action can be defined using the command "set security screen ids-option".

187. The TOE is also capable of detecting the following signatures:

- TCP SYN+RST flags, by defining an custom attack to match "protocol tcp tcp-flags rst" and "protocol tcp tcp-flags syn"[55];
- UDP Chargen DoS attack , by configuring a firewall policy to match the predefined "junos-chargen" with the desired allow/block reaction;
- Flooding of a network (DoS attack), by the configuration of policers that allow establishing prioritization and bandwidth limits for different type of network traffic. (***IPS_SBD_EXT.1.3, IPS_SBD_EXT.1.4***)

188. The TOE allows administrators to define signatures for anomalous traffic in terms of throughput (bits per second), time of the day for defined source/destination address and source/destination port, frequency of traffic patterns and thresholds of traffic patterns.

189. Anomaly signatures based on time of day characteristics are implemented by configuring schedulers using the Junos command 'set schedulers' and attaching them to firewall policies, which in turn specify the target traffic in terms of IP addresses and port numbers as well as the action to be perform on signature triggering (allow or block/drop traffic).

190. Anomaly signatures based on throughput characteristics are implemented by configuring policers with a bandwidth limit and the desired signature action (discard or forward), using the

[55] By default the TOE will drop packets where the TCP flags SYN and ACK are set at the screen level.

Junos command 'set firewall policer', and attaching it to any interface with the Junos command 'set interfaces'. Traffic exceeding the specified throughput limit is dropped when the policer is configured to discard traffic. A policer can be applied to specific inbound or outbound IP packets in a Layer 3 traffic flow at a logical interface by using a stateless firewall filter. If an input firewall filter is configured on the same logical interface as a policer, the policer is executed first. If an output firewall filter is configured on the same logical interface as a policer, the firewall filter is executed first. (***IPS_ABD_EXT.1***)

191. For more information about configuring event logging, see the Junos OS Complete Software Guide for SRX Series Services Gateways (Volume 2), Part 9: 'Intrusion Detection and Prevention Feature Guide for Security Devices' and the Junos OS Common Criteria Evaluated Configuration Guide for SRX Series Security Devices.

192. (***IPS_NTA_EXT.1, IPS_IPB_EXT.1, IPS_SBD_EXT.1, IPS_ABD_EXT.1***)

8 Rationales

8.1 SFR dependency analysis

The dependencies between SFRs implemented by the TOE are satisfied as demonstrated in [NDcPP] Appendix E.1.

Security Functional Requirement	Dependency	Rationale
FAU_GEN.1/ND	FPT_STM.1	FPT_STM_EXT.1 included (which is hierarchic to FPT_STM.1)
FAU_GEN.1/IPS	FPT_STM.1	FPT_STM_EXT.1 included (which is hierarchic to FPT_STM.1)
FAU_GEN.2	FAU_GEN.1 FIA_UID.1	FAU_GEN.1 Included Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator identification timing
FAU_STG_EXT.1	FAU_GEN.1 FTP_ITC.1	FAU_GEN.1 included FTP_ITC.1 included
FAU_STG.1	FAU_GEN.1	FAU_GEN.1 Included
FCS_CKM.1/ND	FCS_CKM.2 or FCS_COP.1 FCS_CKM.4	FCS_CKM.2 included FCS_CKM.4 included
FCS_CKM.1/IKE	FCS_CKM.2 or FCS_COP.1 FCS_CKM.4	FCS_CKM.2 included FCS_CKM.4 included
FCS_CKM.2	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 included (also FTP_ITC.1 as a secure channel that could be used for import) FCS_CKM.4 included
FCS_CKM.4	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1	FCS_CKM.1 included (also FTP_ITC.1 as a secure channel that could be used for import)
FCS_COP.1/DataEncryption	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 included (also FTP_ITC.1 as a secure channel that could be used for import) FCS_CKM.4 included
FCS_COP.1/SigGen	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 included (also FTP_ITC.1 as a secure channel that could be used for import) FCS_CKM.4 included
FCS_COP.1/Hash	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 included (also FTP_ITC.1 as a secure channel that could be used for import) FCS_CKM.4 included
FCS_COP.1/KeyedHash	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 included (also FTP_ITC.1 as a secure channel that could be used for import) FCS_CKM.4 included
FCS_RBG_EXT.1	None	n/a

Security Functional Requirement	Dependency	Rationale
FCS_IPSEC_EXT.1	FCS_CKM.1 FCS_CKM.2 FCS_COP.1/DataEncryption FCS_COP.1/SigGen FCS_COP.1/Hash FCS_COP.1/KeyedHash FCS_RBG_EXT.1	FCS_CKM.1 included FCS_CKM.2 included FCS_COP.1/DataEncryption included FCS_COP.1/SigGen included FCS_COP.1/Hash included FCS_COP.1/KeyedHash included FCS_RBG_EXT.1 included
FCS_SSHS_EXT.1	FCS_CKM.1 FCS_CKM.2 FCS_COP.1/DataEncryption FCS_COP.1/SigGen FCS_COP.1/Hash FCS_COP.1/KeyedHash FCS_RBG_EXT.1	FCS_CKM.1 included FCS_CKM.2 included FCS_COP.1/DataEncryption included FCS_COP.1/SigGen included FCS_COP.1/Hash included FCS_COP.1/KeyedHash included FCS_RBG_EXT.1 included
FIA_AFL.1	FIA_UAU.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FIA_PMG_EXT.1	None	n/a
FIA_UIA_EXT.1	FTA_TAB.1	FTA_TAB.1 included
FIA_UAU_EXT.2	None	n/a
FIA_UAU.7	FIA_UAU.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FIA_X509_EXT.1/Rev	None	n/a
FIA_X509_EXT.2	None	n/a
FIA_X509_EXT.3	FCS_CKM.1 Cryptographic Key Generation	FCS_CKM.1/ND included
FIA_PSK_EXT.1	None	n/a
FMT_MOF.1/ManualUpdate	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MOF.1/Services	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MOF.1/Functions	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MTD.1/CoreData	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MTD.1/CryptoKeys	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_SMF.1/ND	None	n/a
FMT_SMF.1/IPS	None	n/a
FMT_SMR.2	FIA_UID.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FPT_SKP_EXT.1	None	n/a
FPT_APW_EXT.1	None	n/a
FPT_TST_EXT.1	None	n/a
FPT_TST_EXT.3	None	n/a
FPT_TUD_EXT.1	FCS_COP.1/SigGen or FCS_COP.1/Hash	FCS_COP.1/SigGen

Security Functional Requirement	Dependency	Rationale
FPT_STM.EXT.1	None	n/a
FPT_FLS.1/SelfTest	None	n/a
FTA_SSL_EXT.1	FIA_UID.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FTA_SSL.3	None	n/a
FTA_SSL.4	None	n/a
FTA_TAB.1	None	n/a
FTP_ITC.1	None	n/a
FTP_TRP.1/Admin	None	n/a
FDP_RIP.2	None	n/a
FPF_RUL_EXT.1	None	n/a
FFW_RUL_EXT.1	None	n/a
FFW_RUL_EXT.2	None	n/a
IPS_NTA_EXT.1	None	n/a
IPS_IPB_EXT.1	None	n/a
IPS_SBD_EXT.1	None	n/a
IPS_ABD_EXT.1	None	n/a

Table 13 SFR Dependency Analysis

9 Glossary

AES	Advanced Encryption Standard
ANSI	American National Standards Institute
API	Application Program Interface
BGP	Border Gateway Protocol
cPP	collaborative Protection Profile
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CM	Configuration Management
CSP	Critical security parameter
DFA	Deterministic Finite Automaton
DES	Data Encryption Standard
DH	Diffie Hellman
EAL	Evaluation Assurance Level
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
EP	Extended Package, defined in [CC1]
ESP	Encapsulating Security Payload
FFC	Finite Field Cryptography
FIPS	Federal Information Processing Standard
FTP	File Transfer Protocol
FWEP	Firewall Extended Package
HMAC	Keyed-Hash Authentication Code
I&A	Identification and Authentication
ICMP	Internet Control Message Protocol
ID	Identification
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IP	Internet Protocol
IPS	Intrusion Prevention System
IPsec	Internet Protocol Security
IPsec ESP	Internet Protocol Security Encapsulating Security Payload
IPv6	Internet Protocol Version 6
IPX	Internetwork Packet Exchange
ISAKMP	Internet Security Association and Key Management Protocol
IS-IS	Intermediate System-to-Intermediate System
ISO	International Organization for Standardization
IT	Information Technology
JDM	Juniper device manager
JET	Junos Extension toolkit. Control plane APIs for Junos.
Junos	Juniper Operating System
KVM	Kernel-based Virtual Machine
LDP	Label Distribution Protocol
MAC	Mandatory Access Control
MRE	Medium Robustness Environment
NAT	Network Address Translation
NDcPP	Network Device collaborative Protection Profile
NTP	Network Time Protocol
OSI	Open Systems Interconnect
OSP	Organizational Security Policy

OSPF	Open Shortest Path First
PAM	Pluggable Authentication Module
PFE	Packet Forwarding Engine
PFEP	Linux process that manages the PFE. Also referred to as forwarding daemon.
PIC/PIM	Physical Interface Card/Module
PKI	Public Key Infrastructure
PP	Protection Profile
PRNG	Pseudo Random Number Generator
RE	Routing Engine
RFC	Request for Comment
RIP	Routing Information Protocol
RNG	Random Number Generator
RSA	Rivest, Shamir, Adelman
SA	Security Association
SCEP	Simple Certificate Enrollment Protocol
SFP	Small Form-factor Pluggable
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SNMP	Simple Network Management Protocol
SSD	Solid State Drive
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TCP/IP	Transmissions Control Protocol/ Internet Protocol
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF interfaces
UDP	User Datagram Protocol
VM	Virtual Machine
VPN	Virtual Private Network
VPNEP	Virtual Private Network Extended Package

10 TOE Network Interface Options

10.1 SRX1500 Transceivers[56]

Interface Model number	Description
SRX-SFP-1GE-LH	SFP 1000BASE-LH Gigabit Ethernet optic module. 1000BASE-ZX
SRX-SFP-1GE-LX	SFP 1000BASE-LH Gigabit Ethernet optic module. 1000BASE-LX
SRX-SFP-1GE-SX	SFP 1000BASE-SX Gigabit Ethernet optic module. 1000BASE-SX
SRX-SFP-1GE-T	SFP 1000BASE-T Gigabit Ethernet module (uses Cat 5 cable). 1000BASE-T
EX-SFP-10GE-LR	SFP+ 10-Gigabit Ethernet optic module. Meets Extended temperature range requirements. 10GBASE-LR
EX-SFP-10GE-SR	SFP+ 10-Gigabit Ethernet optic module 10GBASE-SR
EX-SFPP-10GE-DAC-1M	10-Gbps full-duplex serial transmission. 10GBASE-DAC (1m max distance)
EX-SFPP-10GE-DAC-3M	10-Gbps full-duplex serial transmission. 10GBASE-DAC (3m max distance)

10.2 SRX4100, SRX 4200 and SRX4600 Transceivers[57]

Interface Model number	Description
EX-SFP-10GE-ER	
SRX-SFP-10GE-LR	SFP+ 10-Gigabit Ethernet optic module. Meets extended temperature range requirements. 10GBASE-LR
SRX-SFP-10GE-SR	SFP+ 10-Gigabit Ethernet optic module. 10GBASE-SR
JNPR-10G-SR-8PACK	SFP+ 10G Base SR Optics 8 pack bundle
SRX-SFP-1GE-LH	SFP 1000BASE-LH Gigabit Ethernet optic module. 1000BASE-ZX
SRX-SFP-1GE-LX	SFP 1000BASE-LH Gigabit Ethernet optic module. 1000BASE-LX
SRX-SFP-1GE-SX	SFP 1000BASE-SX Gigabit Ethernet optic module. 1000BASE-SX
JNPR-1G-SX-8PACK	SFP+ 10G Base SX Optics 8 pack bundle
JNPR-1G-T-8PACK	SFP+ 10G Base-T 8 pack bundle
EX-SFP-1GE-T[58]	SFP 1000BASE-T Gigabit Ethernet module. 1000BASE-T

[56] From SRX Series Services Gateway Transceiver Reference, dated April 2016.
[57] From SRX4100 Services Gateway Hardware Guide, dated 2017-06-26 and SRX4200 Services Gateway Hardware Guide, dated 2019-08-28 , SRX4600 Services Gateway hardware Guide, dated 2019-04-16 .
[58] SRX4100, SRX 4200 and SRX 4600 only support 1000Mbps speed on EX-SFP-1GE-T; 10 Mbps and 100 Mbps speeds are not supported.