Security Target for Junos OS 20.4R1 for SRX345, SRX345-DUAL-AC and SRX380 in Cluster Mode

Juniper Networks, Inc.

Version 1.3

November 25, 2021

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 20.4R1 for SRX345, SRX345-DUAL-AC and SRX380 in Cluster Mode. Cluster Mode is a configuration in where a pair of devices are connected and configured to operate like a single device to provide high availability. This Security Target (ST) is conformant to the requirements of Collaborative Protection Profile for Network Devices v2.1 [NDcPP], and those for Firewall [FW_MOD], Intrusion Prevention Systems [IPS_EP], and VPN Gateway [MOD_VPNGW].

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
[Entropy]	Seeding of the Kernel RBG, Junos OS 20.4R1, The SRX345, 345-DUAL-AC and SRX380, Version 1.0, 2 March 2021
[FW_MOD]	Collaborative Protection Profile Module for Stateful Traffic Filter Firewalls, Version 1.3, 27 September 2019
[FW_MOD_SD]	Evaluation Activities for Stateful Traffic Filter Firewalls PP-Module, September 2019, Version 1.3
[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
[NDcPP-FW]	PP-Configuration for Network Device and Stateful Traffic Filter Firewalls, Version 1.3, 27-September-2019
[NDcPP-FW-VPN]	PP-Configuration for Network Devices, Stateful Traffic Filter Firewalls, and Virtual Private Network (VPN) Gateways, Version: 1.0, 2020-03-06
[SD]	Supporting Document, Evaluation Activities for Network Device cPP, September 2018, version 2.1
[MOD_VPNGW]	PP-Module for Virtual Private Network (VPN) Gateways, version 1.0, 17 September 2019

Product Guide References

[ECG]	Junos OS Common Criteria Guide for SRX345, SRX345-DUAL-AC, and SRX380 in Cluster Mode, Release 20.4R1, Date 01 October 2021

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 for Junos OS 20.4R1 for SRX345, SRX345-DUAL-AC and SRX380 in Cluster Mode
ST Revision	1.3
ST Draft Date	November 25, 2021
Author	Juniper Networks, Inc.
cPP/EP Conformance	[NDcPP], [FW_MOD], [MOD_VPNGW], [IPS_EP]

1.2 TOE Reference

TOE Title	Junos OS 20.4R1 for SRX345, SRX345-DUAL-AC and SRX380 in Cluster Mode

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 a statement of the functional requirements for this TOE
6	Security Assurance Requirements	Contains a statement of the assurance requirements for this TOE
7	TOE Summary Specification	Identifies the IT security functions provided by the TOE and maps them to the applicable security functional 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 ST: 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 *<u>italicized and underlined text</u>*

  e.g. "[selection: *change_default, query, modify, delete, [assignment: other operations*]]" in [CC2] or an ECD might become "<u>change_default</u>, *<u>select_tag</u>*" (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 20.4R1 operating system on Services Gateway appliances SRX345, SRX345-DUAL-AC and SRX380 operated in a Cluster Mode. In cluster mode, a pair of devices are connected together and configured to operate like a single device to provide high availability. When configured as a chassis cluster, the two nodes back up each other, with one node acting as the primary device and the other as the secondary device, ensuring stateful failover of processes and services in the event of system or hardware failure. If the primary device fails, the secondary device takes over processing of traffic. Further details of the Cluster Mode configuration are provided in Sect. 1.6.4.

5. In addition to the high availability achieved through a specific configuration of two instances of the TOE, the Services Gateway appliances primarily support the definition and enforcement of information flow policies among network nodes. The Services Gateway appliances provide stateful inspection of every packet that traverses the network and provide central point of control to manage the network security policy.

6. All information flows from one network node to another pass 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 and that the TOE functions are protected from potential attacks. The TOE also provides tools to manage all security functions.

7. The TOE also provides multi-site virtual private network (VPN) gateway functionality and 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.

1.6 TOE Description

1.6.1 Overview

8. Each Juniper Networks SRX Services Gateway appliance is a security system that supports a variety of high-speed interfaces for medium/large networks and network applications. Juniper Networks routers share common Junos firmware, features, and technology for compatibility across platforms.

9. The appliances are physically self-contained. They house all software, firmware and hardware necessary to perform all functions. The hardware consists of two major 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.

10. 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. The RE also 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.

11. 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.
12. The Services Gateway appliances support numerous routing standards for flexibility and scalability as well as IETF SSHv2 and 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.
13. 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 header and payload data attributes. Anomaly detection based on deviation of the monitored traffic from expected values is also supported.
14. Services Gateway appliances accomplish routing through a Virtual Router (VR) mechanism. The TOE can divide 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.
15. In the evaluated deployment the TOE is managed and configured via Command Line Interface, either via a directly connected console or using SSH connections, which can be further tunnelled over IPsec if required.

1.6.2 Physical boundary

16. The physical boundary of the TOE illustrated in Figure 1 is the entire chassis of the Services Gateway appliance. It includes both the hardware and the firmware of the TOE. The TOE is the Junos OS 20.4R1 firmware running on the SRX345, SRX345-DUAL-AC and SRX380 chassis summarised in Table 2 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. The install package for the appliances is “junos-srxsme-20.4R1.12.tgz”.

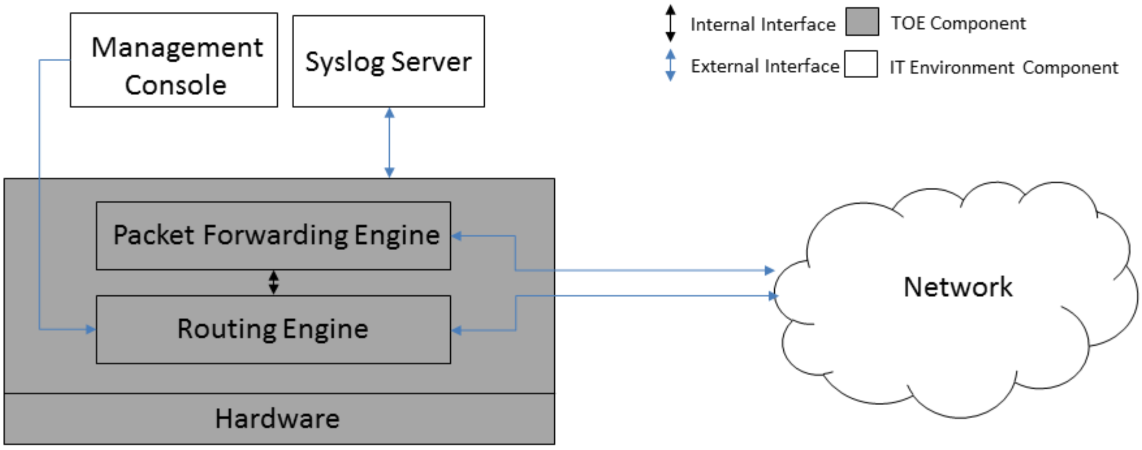

Figure 1 TOE Physical Boundary

17. The TOE interfaces are comprised of the Network interfaces which pass traffic, and Management interface through which the administrative actions are handled.

Model	Network Ports	Firmware
SRX345	• Four Mini PIM (WAN) slots • 8x1Gb Ethernet LAN ports (RJ-45) • 8x1Gb Ethernet LAN ports (SFP) • One console (RJ45 + mini-USB) port • One USB 3.0 port	Junos OS 20.4R1
SRX345-DUAL-AC[1]	• Four Mini PIM (WAN) slots • 8x1Gb Ethernet LAN ports (RJ-45) • 8x1Gb Ethernet LAN ports (SFP) • One console (RJ45 + mini-USB) port • One USB 3.0 port	Junos OS 20.4R1
SRX380	• Four Mini PIM (WAN) slots • 16x1Gb Ethernet LAN ports (RJ-45) • 4x10Gb Ethernet ports (SFP+) • One console (RJ45 + mini-USB) port • One USB 3.0 port	Junos OS 20.4R1

Table 2 TOE Physical Boundary Details

18. Further information about the PIM slots is given in Table 3.

Interface Model number	Description
SRX-MP-1SERIAL-R	Serial Mini-PIM provides physical connection for serial setup
SRX-MP-1T1E1-R	T1/E1 Mini-PIM provides the physical connection to T1 or E1 network media types
SRX-MP1VDSL2-R	VDSL2 Mini-PIM is part of the xDSL family of modem technologies, which provide faster data transmission over a single flat untwisted or twisted pair of copper wires

[1] SRX345-DUAL-AC is a dual power supply version of SRX345. Otherwise, it is identical to SRX345. Identifier SRX345-DUAL-AC is as returned by the user command `show chassis hardware`.

Interface Model number	Description
SRX-MP-LTE-AE (North America, EU) SRX-MP-LTE-AA (Asia, Australia)	Mini-PIM providing wireless WAN support. The Mini-PIM contains an integrated modem and operates over 3G and 4G networks. The Mini-PIM supports up to two SIM cards and can be installed in any of the Mini-PIM slots on the services gateways

Table 3 PIM details

19. The guidance document included as part of the TOE is [ECG].

1.6.3 Logical Boundary

20. The logical boundary of the TOE includes the following security functionality.

Security Functionality	Description
Protected Communications	The TOE implements an SSH server for protected communications between itself and SSH clients. SSH is used by administrators to establish secure sessions between management stations and the TOE and to connect the TOE to external syslog servers. The TOE also implements IPsec for multi-site virtual private network (VPN) gateway functionality and to tunnel remote administrate SSH connections. Each application connecting to the TOE using SSH and IPsec must be successfully authenticated prior to any information exchange. Telnet, File Transfer Protocol (FTP) and Secure Socket Layer (SSL) are out of scope. The TOE includes cryptographic modules which implement the underlying cryptographic services. The cryptographic services include key management and protection of stored keys, cryptographic algorithms, random bit generation, and crypto-administration. The cryptographic modules provide confidentiality and integrity services for authentication and for protecting communications with adjacent systems.
Administrator Authentication	Administrative users must be successfully authenticated using 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 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.
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 5 and Table 6. 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. If the storage limits are reached the oldest logs will be overwritten.

Security Functionality	Description
Management	The TOE provides a Security Administrator role that is authorized and responsible for: 1. configuration and maintenance of cryptographic protocols used in the establishment of secure connections to and from the TOE, 2. regular reviews of all audit data, 3. initiation of trusted update function, 4. administration of VPN, IPS and Firewall functionalities, and 5. all administrative tasks (e.g., creating the security policy). The devices are managed through a Command Line Interface (CLI) which is accessible through a local (serial) console connection or a remote administrative (SSH) session.
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 to each packet. Based on the rules, the TOE determines whether the packet is forwarded or dropped.
Intrusion Prevention	The administrator may configure the TOE to analyze IP-based network traffic forwarded to the TOE's interfaces and detect violations of administrator 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 to interrupt a 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 Cluster Mode configuration

21. The Administrator of the TOE shall set up the Cluster Mode for High Availability (HA) by connecting ge-0/0/1 on node 0 to ge-0/0/1 on node 1. The factory-default configuration does not include HA configuration. To enable HA, if the physical interfaces used by HA have some configurations, these configurations need to be removed. The two hosts constituting a chassis cluster must have identical configuration except for one being configured to node 0 and the other to node 1.

22. The TOE has a dedicated fxp0 interface which is used for the HA management of the TOE. The fxp1 interface for HA control link is ge-0/0/1. The fabric interface fab0 in node 0 and fab1 in node 1 are not fixed and are defined by the Administrator. After the cluster has been defined and set up by the Administrator, the two devices constituting a chassis cluster have identical cluster-id but difference node ID as one host must be node 0 and the other one node 1.

23. The node 1 renumbers its interfaces by adding the total number of system FPCs to the original FPC number of the interface. The renumbering constant for all TOE models is 5 which results in the node 0 interface name remaining ge-0/0/0 and the node 1 interface name being renamed to ge-5/0/0. The fabric interface remains Administrator-defined. An example configuration of the

TOE in Cluster Mode for high availability is given in Figure 2. After clustering, the HA Control link is established between interfaces ge-0/0/1 and ge-5/0/1. The HA fabric link is established between the logical interfaces fab0 and fab1.

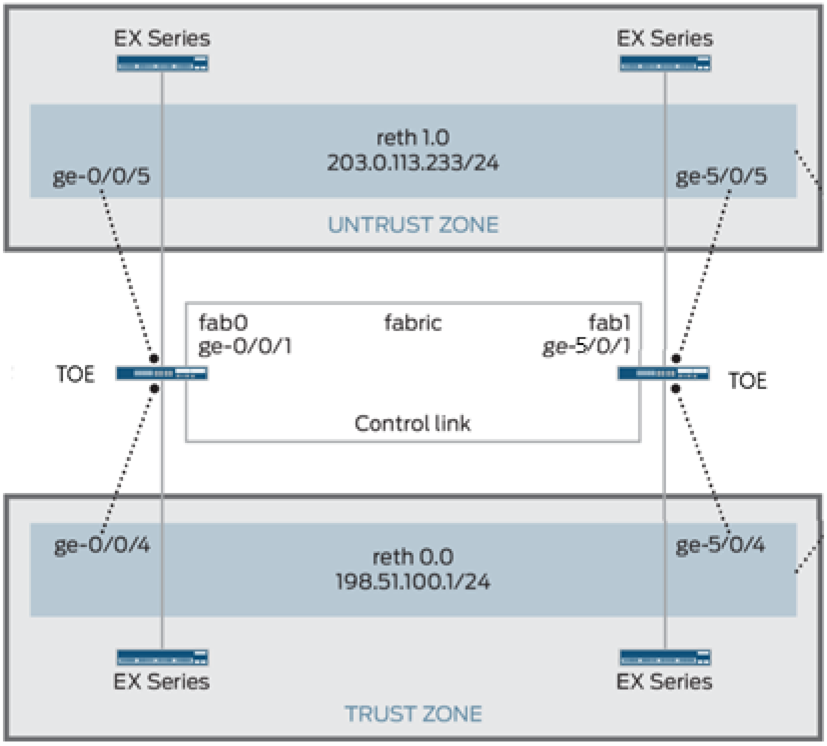

Figure 2 Example of the HA configuration of the TOE

24. Sensitive security configuration data sent over the control link between the two chassis in Cluster Mode is protected using IPsec. Using IPsec for internal communication between devices, the sensitive security configuration information that passes through the chassis cluster link from the primary node to the secondary node is protected from active and passive eavesdropping. Without the internal IPsec key, an attacker cannot gain privilege access or observe traffic.

1.6.5 Non-TOE hardware/software/firmware

25. The TOE requires SFPs/PIMs to operate and communicate with the connected network. The TOE also relies on the provision of the following items, none of which is part of the TOE, 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.

1.6.6 Summary of out scope items

- Use of telnet, since it violates the Trusted Path requirement set (see Section 5.7.2)
- Use of TOE-terminated and TOE-initiated FTP connections, since they violate 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 junos root account.

2 Conformance Claim

2.1 CC Conformance Claim

26. 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

27. This Security Target claims Exact Conformance to [NDcPP] (v2.1). Exact conformance is defined in [NDcPP] section 2 and in [CC_Add].

28. Additionally, as per Sect. 9.6 of [CC1], conformance to PP-Modules is in accordance with specific PP-configurations:

    1. Conformance to [FW_MOD] is claimed in accordance with [NDcPP-FW].
    2. Conformance to [MOD_VPNGW] and [FW_MOD] is claimed in accordance with [NDcPP-FW-VPN].

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

30. The Security Problem definition in this Security Target is consistent with the security problem definitions detailed in the collaborative Protection Profile, collaborative Protection Profile Module, PP-Module and Extended Package to which this ST claims conformance, namely:

    - [NDcPP] Section 4
    - [FW_MOD] Section 4
    - [IPS_EP] Section 2
    - [MOD_VPNGW] Section 3.

31. The statement of the Security Problem Definition in this ST is the union of the threats, organizational security policies and assumptions from the collaborative Protection Profile, collaborative Protection Profile Module, PP-Module and Extended Package.

32. Similarly, the statement of security objectives in this ST is consistent with the statement of security objectives detailed in the collaborative Protection Profile, collaborative Protection Profile Module, PP-Module and Extended Package to which this ST claims conformance, namely:

    - [NDcPP] Section 3
    - [FW_MOD] Section 5
    - [IPS_EP] Section 3
    - [MOD_VPNGW] Section 4.

33. Again, the statement of the Security Objectives in this ST is the union of the security objectives from the collaborative Protection Profile, collaborative Protection Profile Module, PP-Module and Extended Package.

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

    - [NDcPP] Sections 6 & 7

- [FW_MOD] Sections 6 & 7
- [IPS_EP] Section 4 & 6
- [MOD_VPNGW] Section 5.

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

36. This statement of SFRs is augmented with SFRs specified in (at least one of) [NDcPP] Appendix A (Optional requirements), [FW_MOD] Section 6 and Appendices A & B, [IPS_EP] Section 4 and [MOD_VPNGW] Section 5 and Appendices A & B.

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

38. The Security Assurance Requirements specified in this ST are those defined in the base [NDcPP].

39. 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

40. 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
TD0572	NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	FTP_ITC.1	2021.01.29	Yes
TD0571	NiT Technical Decision for Guidance on how to handle FIA_AFL.1	FIA_UAU.1, FIA_PMG_EXT.1	2021.01.29	Yes
TD0570	NiT Technical Decision for Clarification about FIA_AFL.1	FIA_AFL.1	2021.01.29	Yes
TD0549	Consistency of Security Problem Definition update for MOD_VPNGW_v1.0 and MOD_VPNGW_v1.1	Sect. 6.1.2	2020.10.02	Yes
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	AVA_VAN.1	2020.10.15	Yes
TD0545	NIT Technical Decision for Conflicting FW rules cannot be configured (extension of RfI#201837)	FFW_RUL_EXT.1.8	2020.10.15	Yes
TD0538	NIT Technical Decision for Outdated link to allowed-with list	Section 2	2020.07.13	Yes
TD0536	NIT Technical Decision for Update Verification Inconsistency	AGD_OPE.1	2020.07.13	Yes
TD0528	NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	FCS_NTP_EXT.1.4	2020.07.13	No
TD0325	Inline mode for Signature-based IPS policies	IPS_SBD_EXT.1.5	2018.05.21	Yes

Table 4 Applicable NIAP Technical Decisions

41. 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

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

3.1 Threats

43. The following threats for this TOE are as defined in [NDcPP] Section 4.1, which also apply to [FW_MOD], [MOD_VPNGW] and [IPS_EP]. 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

  An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

44. The following additional threats specified in [FW_MOD], [IPS_EP] and [MOD_VPNGW] are also detailed for this TOE:

- T.NETWORK_DISCLOSURE[2]

  Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to conduct unauthorized activities. 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 (e.g., as a result of a phishing episode or by inadvertent responses to email messages), then those internal devices may be susceptible to the unauthorized disclosure of information.

  From an infiltration perspective, VPN gateways serve not only to limit access to only specific destination network addresses and ports within a protected network, but whether network traffic will be encrypted or transmitted in plaintext. With these limits, general network port scanning can be prevented from reaching protected networks or machines, and access to information on a protected network can be limited to that obtainable from specifically configured ports on identified network nodes (e.g., web pages from a designated corporate web server). Additionally, access can be limited to only specific source addresses and ports so that specific networks or network nodes can be blocked from accessing a protected network thereby further limiting the potential disclosure of information.

  From an exfiltration perspective, VPN gateways serve to limit how network nodes operating on a protected network can connect to and communicate with other networks limiting how and where they can disseminate information. Specific external networks can be blocked altogether or egress could be limited to specific addresses and/or ports. Alternately, egress options available to network nodes on a protected network can be carefully managed in order to, for example, ensure that outgoing

[2] Wording from [MOD_VPNGW]

connections are encrypted to further mitigate inappropriate disclosure of data through packet sniffing.

- T. NETWORK_ACCESS[3]

  Devices located outside the protected network may seek to exercise services located on the protected network that are intended to only be accessed from inside the protected network or only accessed by entities using an authenticated path into the protected network. Devices located outside the protected network may, likewise, offer services that are inappropriate for access from within the protected network.

  From an ingress perspective, VPN gateways can be configured so that only those network servers intended for external consumption by entities operating on a trusted network (e.g., machines operating on a network where the peer VPN gateways are supporting the connection) are accessible and only via the intended ports. This serves to mitigate the potential for network entities outside a protected network to access network servers or services intended only for consumption or access inside a protected network.

  From an egress perspective, VPN gateways can be configured so that only specific external services (e.g., based on destination port) can be accessed from within a protected network, or moreover are accessed via an encrypted channel. For example, access to external mail services can be blocked to enforce corporate policies against accessing uncontrolled e-mail servers, or, that access to the mail server must be done over an encrypted link.

- T.NETWORK_MISUSE[4]

  Devices located outside the protected network, while permitted to access particular public services offered inside the protected network, may attempt to conduct inappropriate activities while communicating with those allowed public services. Certain services offered from within a protected network may also represent a risk when accessed from outside the protected network.

  From an ingress perspective, it is generally assumed that entities operating on external networks are not bound by the use policies for a given protected network. Nonetheless, VPN gateways can log policy violations that might indicate violation of publicized usage statements for publicly available services.

  From an egress perspective, VPN gateways can be configured to help enforce and monitor protected network use policies. As explained in the other threats, a VPN gateway can serve to limit dissemination of data, access to external servers, and even disruption of services –all of these could be related to the use policies of a protected network and as such are subject in some regards to enforcement. Additionally, VPN gateways can be configured to log network usages that cross between protected and external networks and as a result can serve to identify potential usage policy violations

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

- T.MALICIOUS_TRAFFIC

  An attacker may attempt to send malformed packets to a machine in hopes of causing the network stack or services listening on UDP/TCP ports of the target machine to crash.

[3] Wording from [MOD_VPNGW]
[4] Wording from [MOD_VPNGW]

46. 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.

47. The following threats specified in [MOD_VPNGW] 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 communicate with those external devices then the data contained within the communications may be susceptible to a loss of integrity.

- 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.

48. No threats are identified for this TOE in addition to those specified in the collaborative Protection Profile, PP Modules and Extended Package.

3.2 Assumptions

49. The assumptions made for this TOE are as defined in [NDcPP] Section 4.2, 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.

- A.RESIDUAL_INFORMATION

  The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

- A.NO_THRU_TRAFFIC_PROTECTION

  This assumption is only applicable to interfaces in the TOE that are defined by the [NDcPP]. For these interfaces, the TOE does not provide any assurance regarding the protection of traffic that traverses it.

50. The following assumption A.CONNECTIONS is introduced through compliance to [MOD_VPNGW] 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.

- 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.

51. No assumptions are identified for this TOE in addition to those specified in the collaborative Protection Profile, PP-modules and Extended Packages.

3.3 Organizational Security Policies

52. The OSP P.ACCESS_BANNER applied for this TOE is as defined in [NDcPP] Section 4.3. The OSP P.ANALYZE applied for this TOE is as defined in [IPS_EP] 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

53. 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

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

55. These are augmented by the statement of security objectives for the TOE in relation to the IPS capabilities as detailed in [IPS_EP] Table A-5, 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_REACT

  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

  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

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

56. These are further augmented by the statement of security objectives for the TOE in relation to the Packet Filtering capabilities as detailed in [FW_MOD] Section 5.1, namely

- O.RESIDUAL_INFORMATION

  The TOE shall implement measures to ensure that any previous information content of network packets sent through the TOE is made unavailable either upon deallocation of the memory area containing the network packet or upon allocation of a memory area for a newly arriving network packet or both.

- O.STATEFUL_TRAFFIC_FILTERING

  The TOE shall perform stateful traffic filtering on network packets that it processes. For this the TOE shall support the definition of stateful traffic filtering rules that allow to permit or drop network packets. The TOE shall support assignment of the stateful traffic filtering rules to each distinct network interface. The TOE shall support the processing of the applicable stateful traffic filtering rules in an administratively defined order. The TOE shall deny the flow of network packets if no matching stateful traffic filtering rule is identified.

  Depending on the implementation, the TOE might support the stateful traffic filtering of Dynamic Protocols (optional).

57. These are further augmented by the statement of security objectives for the TOE in relation to the IPS capabilities as detailed in [MOD_VPNGW] 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.AUTHENTICATION – as also defined by the inverse of the threats defined in [NDcPP] Section 4.1

  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 and ensure that any such connection attempt is both authenticated and authorized. VPN endpoints authenticate each other to ensure they are communicating with an authorized external IT entity.

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

  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 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.SYSTEM_MONITORING – as also defined by the inverse of the threats defined in [NDcPP] Section 4.1

  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

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

58. The statement of security objectives for the operational environment of this TOE is as defined in [NDcPP] Section 5.1, [MOD_VPNGW] Section 4.2 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.

- 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.RESIDUAL_INFORMATION

  The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

- OE.CONNECTIONS

  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.

- OE.NO_THRU_TRAFFIC_PROTECTION

  Except for interfaces covered by the FW_MOD or MOD_VPNGW, the TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

4.3 Security Objectives rationale

59. As these objectives for the TOE and operational environment are the same as those specified in [NDcPP], [FW_MOD], [MOD_VPNGW] 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 Profile, PP Modules and Extended Package to which this ST claims conformance

- [NDcPP] section 4
- [FW_MOD], section 5.3
- [IPS_EP] section 3 & Annex A.2
- [MOD_VPNGW]and section 4.3.

5 Security Functional Requirements

60. All security functional requirements are taken from the [NDcPP] collaborative Protection Profile and from the [FW_MOD], [IPS_EP] and [MOD_VPNGW] Extended Packages. The Security Functional requirements are primarily structured according to [NDcPP], with requirements and operations from [FW_MOD], [IPS_EP] and [MOD_VPNGW] 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.

61. 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>*
     - *<u>Cluster Mode configuration and management,</u>*
     - *<u>synchronization of the kernel state between the two Routing Engines configured in Cluster Mode.</u>*

     ];
d) *Specifically defined auditable events listed in Table 5.*

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 5*.

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG_EXT.1	None	None
FCS_CKM.1	None	None
FCS_CKM.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/IPS	None	None
FMT_SMF.1/ND	None	None
FMT_SMF.1/FFW	All management activities of TSF data (including creation, modification and deletion of firewall rules).	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_EXT.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. Termination of the trusted path. Failure of the trusted path functions.	None.
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate. Any addition, replacement or removal of trust anchors in the TOE's trust store.	Reason for failure of certificate validation. Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store.
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
FFW_RUL_EXT.2	Dynamical definition of rule Establishment of a session	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

Table 5 FAU_GEN.1 Security Functional Requirements and Auditable Events[5]

FAU_GEN.1/IPS IPS Audit Data Generation[6]

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 Table 6*].

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.

[5] Includes also audit evens from [MOD_VPNGW] and [FW_MOD].
[6] 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 6 Audit Events and Details from [IPS_EP]

5.1.1.2 FAU_GEN.2 User identity association

FAU_GEN.2 User identity association

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

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

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 The TSF shall be able to store generated audit data on the TOE itself. [TOE shall consist of a single standalone component that stores audit data locally.

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

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

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

- *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;*
- *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;*
- *FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3*[7]

] ~~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)[8]

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

- **FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes;**
- **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]]**

and

- **no other key generation algorithms]**

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: *[*

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

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

[7] As per Network Device Interpretations #201723rev2, in NITDecisionRfi201723rev2.pdf
[8] In accordance with [MOD_VPNGW].

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 [*
    - *logically addresses the storage location of the key and performs a [single overwrite consisting of [zeroes]]*

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)[9]

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

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, 521 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*.

[9] As per [MOD_VPNGW]

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 and 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 [*[1]* 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), AES-GCM-128, AES-GCM-256 (specified in RFC 4106)] and [AES-CBC-192 (specified in RFC 3602), AES-GCM-192 (specified in RFC 4106)] together with a Secure Hash Algorithm (SHA)-based HMAC [HMAC-SHA-1, HMAC-SHA-256][10].

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 [

[10] In accordance with [MOD_VPNGW]

  - length of time, where the time values can be configured within [*0.2-24*] hours[11];

  ];
- 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 [*224 (for DH Group 14), 256 (for DH Groups 19 and 24) and 384 (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) 19 (256-bit Random ECP), 20 (384-bit random ECP) and [14 (2048-bit MODP), 24 (2048-bit MODP with 256-bit POS)][12].

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)** [*SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN*][13].

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, 4344, 5656, 6668].

[11] Length of time can be configured between 180 seconds and 86,400 seconds.
[12] In accordance with [MOD_VPNGW].
[13] In accordance with [MOD_VPNGW].

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].

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*].

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.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses *[hmac-sha1, hmac-sha2-256, hmac-sha2-512]* as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that *[diffie-hellman-group14-sha1, ecdh-sha2-nistp256]* and *[ecdh-sha2-nistp384, ecdh-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 each encryption key is used to protect no more than one gigabyte of data. After any 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

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*.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met, the TSF shall *[prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until [Security Administrator has unlocked the account] is taken by an Administrator; prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until an Administrator defined time period has elapsed]*.

ST Application Note
The Security Administrator can select to unlock the account of another administrator who has failed to authenticate through the local console, 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

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

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[14]

FIA_UAU_EXT.2.1 The TSF shall provide a local **password-based** authentication mechanism to perform local administrative user authentication.

5.3.4.2 FIA_UAU.7 Protected Authentication Feedback

FIA_UAU.7 Protected Authentication Feedback

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

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 designated as a trust anchor..
- 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.
- The TSF shall validate the revocation status of the certificate using [a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3]
- 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.*

[14] In accordance with TD408.

- *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 IPSec and [*no other protocols*], and [*no additional uses*].[15]

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

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 as specified by RFC 2986 and be able to provide the following information in the request: public key and [device-specific information, Common Name, Organization, Organizational Unit, Country].[16]

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

5.3.5.4 FIA_PSK_EXT.1 Pre-Shared Keys

5.3.5.5 FIA_PSK_EXT.1 Pre-Shared Keys [17]

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

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.

[15] In accordance with [MOD_VPNGW]
[16] in accordance with [MOD_VPNGW]
[17] In accordance with [MOD_VPNGW].

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 **start and stop** ~~**the functions**~~ **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*.

5.4.2 Management of TSF Data (FMT_MTD)

5.4.2.1 *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*.

5.4.2.2 *FMT_MTD.1/CryptoKeys Management of TSF data*

FMT_MTD.1/CryptoKeys Management of TSF data[18]

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[19]

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 manage cryptographic keys****;*

[18] In accordance with [MOD_VPNGW]
[19] In accordance with [MOD_VPNGW]

- ***Ability to configure the cryptographic functionality;***
- ***Ability to configure the lifetime for IPsec SAs;***
- ***Ability to import X.509v3 certificates to the TOE's trust store;***
- ***Ability to enable, disable, determine and modify the behavior of all the security functions of the TOE identified in ~~this PP Module~~ [MOD_VPNGW];***
- ***Ability to configure all security management functions identified in [MOD_VPNGW]~~other sections of this PP Module~~;***

  [
    - *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[20]

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

- *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*]

FMT_SMF.1/FFW Specification of Management Functions[21]

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

- *Ability to configure firewall rules*[22]

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:

[20] In accordance with [IPS_EP].
[21] From [FW_MOD]
[22] For conformance with [FW_MOD], the statement of the SFR does not include the square brackets as are used in the statements of other iterations of FMT_SMF.1.

- *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 administrative passwords in non-plaintext form.

FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext administrative 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[23]

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [<u>during initial start-up (on power on)</u>] to demonstrate the correct operation of the TSF: **noise source health tests**, [

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

5.5.3.2 FPT_TST_EXT.3 TSF Self-Test with Defined Methods

FPT_TST_EXT.3 TSF Testing[24]

FPT_TST_EXT.3.1 The TSF shall run a suite of the following self-tests [[when loaded for execution]] to demonstrate the correct operation of the TSF: [integrity verification of stored executable code].

FPT_TST_EXT.3.2 The TSF shall execute the self-testing through [a TSF-provided cryptographic service specified in FCS_COP.1/SigGen].

[23] In accordance with [MOD_VPNGW]

[24] In accordance with [MOD_VPNGW].

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[25].

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].

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 [26]

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

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, [*terminate the session]* 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

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

[25] In accordance with [MOD_VPNGW].
[26] In accordance with [MOD_VPNGW].

5.6.2.2 FTA_SSL.4 User-initiated Termination (Refinement)

FTA_SSL.4 User-initiated Termination

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

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.

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

FTP_ITC.1.1 The TSF shall use **[<u>IPsec</u>, <u>SSH</u>]** to provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [<u>node configured in Cluster Mode</u>]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data **from disclosure and detection of modification of the channel data.**

FTP_ITC.1.2 The TSF shall permit **<u>the</u> <u>TSF or the authorized IT entities</u>** to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [*streaming of syslog events, communication between two hosts configured in Cluster Mode*].

FTP_ITC.1/VPN Inter-TSF trusted channel (VPN Communications)[27]

FTP_ITC.1.1/VPN The TSF shall **be capable of using IPsec to** provide a communication channel between itself and **authorized IT entities supporting VPN communications** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **disclosure and detection of modification of the channel data**.

FTP_ITC.1.2/VPN The TSF shall permit [the authorized IT entities] to initiate communication via the trusted channel.

FTP_ITC.1.3/VPN The TSF shall initiate communication via the trusted channel for [*remote VPN gateways/peers*].

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 [<u>SSH, IPsec</u>] to** provide a communication path between itself and **authorized** <u>remote</u> **Administrators** that is logically distinct from other

[27] In accordance with [MOD_VPNGW].

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[28]

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.

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[29]

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 allow the definition of Packet Filtering rules using the following network protocol fields:

- IPv4 (RFC791)
  - Source address
  - Destination Address
  - Protocol
- IPv6 (RFC2460)
  - Source address
  - Destination Address
  - Next Header (Protocol)
- TCP (RFC793)
  - Source Port
  - Destination Port
- UDP (RFC768)
  - Source Port
  - Destination Port

FPF_RUL_EXT.1.3 The TSF shall allow the following operations to be associated with Packet Filtering rules: permit and drop with the capability to log the operation.

FPF_RUL_EXT.1.4 The TSF shall allow the Packet Filtering rules to be assigned to each distinct network interface.

[28] In accordance with [FW_MOD].
[29] In accordance with [MOD_VPNGW].

FPF_RUL_EXT.1.5 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.1.6 The TSF shall drop traffic if a matching rule is not identified.

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[30]

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;

[30] From [FW_MOD]

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[31]

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[32]

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*];
    - Session-reset-capable interfaces: [*Ethernet interfaces*];
    - 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[33]

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].

[31] In accordance with [FW_MOD].

[32] In accordance with [IPS_EP].

[33] In accordance with [IPS_EP].

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[34]

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

[34] In accordance with [IPS_EP].

    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 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][35]

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[36]

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].

[35] In accordance with TD0325

[36] In accordance with [IPS_EP].

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: [
  - <u>allow the traffic flow;</u>
  - <u>send a send a TCP reset to the source address of the offending traffic;</u>
  - <u>send a TCP reset to the destination address of the offending traffic]</u>
- In inline mode:
  - allow the traffic flow
  - block/drop the traffic flow
- and [<u>no other actions</u>]

6 Security Assurance Requirements

62. The TOE security assurance requirements are taken from [NDcPP] , together with the refinements documented in [NDcPP] Section 7, as listed in Table 7 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)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Basic functional specification (ADV_FSP.1)
Guidance documents (AGD)	Preparative procedures (AGD_PRE.1)
	Operational user guidance (AGD_OPE.1)
Life cycle support (ALC)	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 7 Security Assurance Requirements

7 TOE Summary Specification

7.1 Protected communications

63. 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

64. All FIPS-approved cryptographic functions implemented by the secure network appliance are implemented in the following libraries:

- Junos OS 20.4R1 – Dataplane (for IPsec for customer ports)
- Junos OS 20.4R1 – Quicksec (for IKE Daemon for both customer ports and HA control link)
- Junos OS 20.4R1 - OpenSSL (for SSH and the PKI daemons, and DRBG for all daemons) in Control Plane
- Junos OS 20.4R1 - OpenSSH (for SSH Daemon)
- Junos OS 20.4R1 – LibMD (MGD daemon)
- Junos OS 20.4R1 - Kernel (for veriexec and IPsec for HA control link)

65. 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.

66. **The network appliance is to be operated with FIPS mode enabled.**

67. 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 8.

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	SFRs Supported	Certificate Number	SRX345 and SRX345-DUAL-AC Processor	SRX380 Processor
SPU: Data Plane	Data plane – IPSec Daemon	FIPS 197, SP 800-38D	AES-GCM (128, 192, 256) (Encrypt, Decrypt, AEAD)	FCS_COP.1/DataEncryption	A1502	Cavium Octeon III (CN7130) w/ internal accelerators (HW/FW)	Cavium Octeon II (CN7360) w/ internal accelerators (HW/FW)
		FIPS 197, SP 800-38A	AES-CBC (128, 192, 256) (Encrypt, Decrypt)	FCS_COP.1/DataEncryption			
		FIPS 180-4	SHS: SHA (1, 256) Byte Oriented (Message Digest Generation)	FCS_COP.1/Hash			
		FIPS 198-1	HMAC-SHA (1, 256) Byte Oriented (Message Authentication)	FCS_COP.1/KeyedHash			
SPU: Control Plane	Quicksec – IKED Daemon	FIPS 197, SP 800-38A	AES-CBC (128, 192, 256) (Encrypt, Decrypt)	FCS_COP.1/DataEncryption	A1501	Cavium Octeon III (CN7130) (FW)	Cavium Octeon II (CN7360) (FW)
		FIPS 197, SP 800-38D	AES-GCM (128, 256) (Encrypt, Decrypt, AEAD)	FCS_COP.1/DataEncryption			
		FIPS 180-4	SHS: SHA (256, 384) Byte Oriented (Message Digest Generation)	FCS_COP.1/Hash			
		FIPS 198-1	HMAC-SHA (256, 384)	FCS_COP.1/KeyedHash			
		SP 800-135	IKE v1/v2 KDF (SHA-256, SHA-384)	FCS.CKM.2			

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	SFRs Supported	Certificate Number	SRX345 and SRX345-DUAL-AC Processor	SRX380 Processor
RE: Control Plane	Quicksec – All Daemons	SP 800-90A	DRBG (HMAC-SHA-256) (Random Bit Generation)	FCS_RBG_EXT.1	A1501	Cavium Octeon III (CN7130) (FW)	Cavium Octeon II (CN7360) (FW)
SPU: Control Plane	Octeon – IKED Daemon	FIPS 186-4	RSA PKCS1_V1_5 (n=2048 (SHA 256), n=4096 (SHA 256)) (SigGen, SigVer)	FCS_COP.1/SigGen	A1501	Cavium Octeon III (CN7130) w/ internal accelerators (HW/FW) - Partly accelerated	Cavium Octeon II (CN7360) w/ internal accelerators (HW/FW) - Partly accelerated
		FIPS 186-4	ECDSA (P-256 w/ SHA-256) ECDSA (P-384 w/ SHA-384) (SigGen, SigVer, KeyGen for ECDH)	FCS_COP.1/SigGen FCS.CKM.1/IKE FCS.CKM.2			
RE: Control Plane	Octeon – IKED Daemon	SP 800-56A Rev. 3	KAS-SSC (Group 14, 19, 20, 24).	FCS.CKM.2	A1502	Cavium Octeon III (CN7130) w/ internal accelerators (HW/FW) - Partly accelerated	Cavium Octeon II (CN7360) w/ internal accelerators (HW/FW) - Partly accelerated
RE: Control Plane	OpenSSL – SSHD Daemon and PKID Daemon	FIPS 197, SP800-38A	AES-CBC/CTR (128, 192, 256) (Encrypt, Decrypt)	FCS_COP.1/DataEncryption	A1836	Cavium Octeon III (CN7130) (FW)	Cavium Octeon II (CN7360) (FW)
		FIPS 180-4	SHS: SHA (1, 256, 384, 512) Byte Oriented (Message Digest Generation, KDF Primitive)	FCS_COP.1/Hash			

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	SFRs Supported	Certificate Number	SRX345 and SRX345-DUAL-AC Processor	SRX380 Processor
		FIPS 198-1	HMAC-SHA (1, 256, 512) Byte Oriented (Message Authentication DRBG Primitive)	FCS_COP.1/KeyedHash			
		FIPS 186-4	RSA PKCS1_V1_5 (n=2048 (SHA-256), n=4096 (SHA-256)) (SigGen, SigVer)	FCS_COP.1/SigGen			
		FIPS 186-4	RSA X931 (n=2048 (SHA-256), n=4096 (SHA-256)) (KeyGen)	FCS_CKM.1/ND FCS_CKM.1/IKE			
		FIPS 186-4	ECDSA [P-256 (SHA-256)], [P-384 (SHA-384)], [P-521 (SHA-512)] (SigGen, SigVer, KeyGen)	FCS_COP.1/SigGen FCS_CKM.1/ND			
		SP 800-56 A Rev. 3	KAS-SSC (Group 14, 19, 20, 21)	FCS_CKM.2			
RE: Control Plane	OpenSSL – All Daemons	SP 800-90A	DRBG (HMAC-SHA-2-256) (Random Bit Generation)	FCS_RBG_EXT.1	A1836	Cavium Octeon III (CN7130) (FW)	Cavium Octeon II (CN7360) (FW)

Module	Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	SFRs Supported	Certificate Number	SRX345 and SRX345-DUAL-AC Processor	SRX380 Processor
RE: Control Plane	OpenSSH – SSHD Daemon	SP 800-135	SSH v2 KDF (SHA 1, SHA-256, SHA-384) (Key Derivation)	FCS_CKM.2	A1505	Cavium Octeon III (CN7130) (FW)	Cavium Octeon II (CN7360) (FW)
RE: Control Plane	Libmd - MGD Daemon, Password Hashing	FIPS 180-4	SHS: SHA (1, 256) Byte Oriented (Message Digest Generation)	FCS_COP.1/Hash	A1504	Cavium Octeon III (CN7130) (FW)	Cavium Octeon II (CN7360) (FW)
		FIPS 198-1	HMAC-SHA (256) Byte Oriented Message Digest Generation)	FCS_COP.1/KeyedHash			
RE: Control Plane	Kernel – Veriexec, kernel DRBG, IPsec for HA control link	SP800-90A	DRBG (HMAC-SHA-2-256) (Random Bit Generation)	FCS_RBG_EXT.1	A1503	Cavium Octeon III (CN7130) (FW)	Cavium Octeon II (CN7360) (FW)
		FIPS 198-1	HMAC-SHA (1, 256) Byte Oriented (Message Authentication, DRBG Primitive)	FCS_COP.1/KeyedHash			
		FIPS 180-4	SHS: SHA (1, 256) Byte Oriented (Message Digest Generation)	FCS_COP.1/Hash			
		FIPS 197, SP 800-38A	AES-CBC (128, 192, 256) (Encrypt, Decrypt)	FCS_COP.1/DataEncryption			

Table 8 CAVP Certificate Results for Cryptographic Services

68. 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***)

69. 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***).

70. 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 9: 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.

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 AES GCM 128 AES GCM 256	HMAC-SHA-256 HMAC-SHA-384
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 256	HMAC-SHA-256 HMAC-SHA-384
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 AES GCM 128 AES GCM 192 AES GCM 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	ECDSA P-256 ECDSA P-384 ECDSA P-521 ssh_rsa	AES CTR 128 AES CTR 256 AES CBC 128 AES CBC 256	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512

Table 9 Supported Protocols

71. The TOE acts as both sender and recipient for IPsec and only as the server for SSH in the supported protocols listed in Table 9. The TOE implements Diffie-Hellman group 14, using the modulus and generator specified by Section 3 of RFC3526. (***FCS_CKM.2***)

72. Zeroization of the critical security parameters is handled as stated in Table 10 (***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 SDD)	When the appliance is recommissioned, the config files (including CSP files such as SSH keys) are removed using the “request system zeroize” option.
	Loaded into memory to complete session establishment	Plaintext	Memory	Memory free() operation is performed by Junos upon session termination
SSH Session Key	Session keys used with SSH, AES 128, 256, hmac-sha-1 or hmac-sha2-256, hmac-sha2-512, DH Private Key (2048 or elliptic curve 256/384/521-bits)	Plaintext	Memory	Memory free() operation is performed by Junos upon session termination
User Password	Plaintext value as entered by user	Plaintext as entered	Processed in Memory	Memory free() operation is performed by Junos upon completion of authentication
		Hashed when stored (HMAC-sha1, sha256, sha512)	Stored on disk	When the appliance is recommissioned, the config files (including the obfuscated password) are removed using the “request system zeroize” 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, RSA 4096, ECDSA P-256, ECDSA P-384	Plaintext	Disk/Memory	‘clear security ike security-association’ command (‘clear security IKE security-association ha-link-encryption’ for the HA control link tunnel) or reboot the box. Private keys stored in flash are not zeroized unless an explicit “request system zeroize” is executed.

CSP	Description	Method of storage	Storage location	Zeroization Method
IKE-SKEYID	IKE master secret used to derive IKE and IPsec ESP session keys	Plaintext	Memory	‘clear security ike security-association’ command (‘clear security IKE security-association ha-link-encryption’ for the HA control link tunnel) or reboot the box.
IKE Session Keys	IKE session key. AES, HMAC	Plaintext	Memory	‘clear security ike security-association’ command (‘clear security IKE security-association ha-link-encryption’ for the HA control link tunnel) or reboot the box.
ESP Session Key	ESP session keys. AES, HMAC	Plaintext	Memory	‘clear security ipsec security-association’ command (‘clear security IKE security-association ha-link-encryption’ for the HA control link tunnel) 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 (‘clear security IKE security-association ha-link-encryption’ for the HA control link tunnel) or reboot the box.
IKE-PSK	Pre-shared authentication key used in IKE.	Encrypted	Disk/Memory	‘clear security IKE security-association’ command (‘clear security IKE security-association ha-link-encryption’ for the HA control link tunnel) 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

Table 10 CSP Storage and Zeroization

73. 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[37]. Encrypted or obfuscated passwords can be viewed by Security Administrators using the CLI command 'request system decrypt password'. (***FPT_SKP_EXT.1***)

7.1.2 Random Bit Generation

74. Junos OS performs random bit generation in accordance with NIST Special Publication 800-90 using HMAC_DRBG, SHA-256. The RBG is seeded from both software and hardware sources. The TOE relies on RANDOM_NET and RANDOM_PURE (Cavium Oscillators) as sources of entropy. Details are given in [Entropy].

7.1.3 SSH

75. 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***)

76. 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***)

77. 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***)

78. The Junos OS SSH server is implemented in accordance with RFCs 4251, 4252, 4253, 4254, 4344 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:

[37] Security Administrators do not have root permission in shell.

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. It 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. 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.
RFC 4344	The Secure Shell (SSH) Transport Layer Encryption Modes	**Encryption modes:** The TOE implements the following encryption methods: aes128-ctr and aes256-ctr.
RFC 5656	SSH ECC Algorithm Integration	**ECDH Key Exchange**: The support key exchange methods specified in this RFC are ecdh-sha2-nistp256 or ecdh-sha2-nistp384 and 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 algorithm, provided it has a message digest size of 256 bits.**Required Curves:** All required curves are implemented: ecdh-sha2-nistp256, ecdh-sha2-nistp384 and 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 algorithm hmac-sha2-256 and hmac-sha2-512 are implemented for SSH transport.

Table 11 SSH RFC conformance

79. 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

80. The TOE is conformant to RFC 4301 (***FCS_IPSEC_EXT.1.1***) and supports IPsec in tunnel mode only (***FCS_IPSEC_EXT.1.3***). IPsec is used for VPN communications between the TOE and IPsec peers (***FCS_ITC.1/VPN***), to protect audit log data between the TOE and the audit server, and to protect security sensitive configuration data exchanged between two instances of the TOE configured in Cluster Mode via the HA control link (***FTP_ITC.1***). IPsec can also be used for tunnelling the SSH traffic in the establishment of a trusted path for authenticating the Administrator of the TOE (**FTP_TRP.1/Admin**). There is a single IKE daemon, which is used to negotiate all IPsec tunnels; however there are two implementations of IPSec: one for customer VPN communications implemented in the data plane, and one for the HA control link tunnel implemented in the control plane kernel.

81. By default, the TOE denies all traffic through an SRX Series device. 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.

82. 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

83. 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. In the case of the IPsec for the HA control link, the security policy rule is not configurable. The TOE does not route any traffic through a HA link. The only data sent through the HA control link IPsec tunnel is security sensitive configuration data between the TOE to the other Cluster node. Security sensitive configuration data consists of the following:

  - contents of the configuration file
  - rcp and rsh data between the nodes
  - IPsec security association data for established VPN customer tunnels

(***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***)

84. 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. For the HA control link tunnel, the TOE restricts its support of encryption algorithm to AES-CBC-128, AES-CBC-192 or AES-CBC-256. (***FCS_IPSEC_EXT.1.4***)

85. 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. The HA control link tunnel only uses IKEv2. (***FCS_IPSEC_EXT.1.5***)

86. 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***)

87. In the evaluated configuration, the TOE permits configuration of the IKE and IPsec lifetime exchanges for customer VPN tunnels in terms of number of (kilo)bytes (64 to 4294967294 kilo bytes) or length of time (180 to 86400 seconds). The TOE does not allow users to configure the IPsec lifetime (in terms of (kilo) bytes) of the HA control link tunnel. (***FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8***)

88. 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>
```

89. 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 if no acceptable match is found. (***FCS_IPSEC_EXT.1.11***)

90. 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***)

91. 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***)

92. 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***)

93. The TOE uses pre-shared keys for IPSec. The TOE accepts ASCII pre-shared or bit-based keys of 1 to 255 characters (and their binary equivalent) that 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).

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

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

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

96. 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***)

97. 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).

98. 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.

99. 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.

100. 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***)

101. 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.

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

103. 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***

104. 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

105. Junos OS enforces binding between human users and subjects. The Security Administrator[38] is responsible for provisioning user accounts, and only the Security Administrator can do so. (***FMT_SMR.2***)

106. 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.

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

- login

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

- PAM Library module

108. 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.

109. 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).

110. 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.

111. 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 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.

112. 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.

113. 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.

114. 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***)

115. 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.

116. 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***)

117. User sessions (local and remote) can be terminated by users (***FTA_SSL.4***). The administrative user can logout of the existing session by typing exit 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.

118. 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.

119. 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

120. Junos OS runs the following set of self-tests during power on to check the correct operation of the Junos OS firmware (***FPT_TST_EXT.1***):

- Power on test – determines the boot-device responds, and performs a memory size check to confirm the amount of available memory.
- File integrity test –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.
- Crypto integrity test – checks integrity of major CSPs, such as SSH hostkeys and iked credentials, such as Cas, CERTS, and various keys.
- Authentication error – verifies that veriexec is enabled and operates as expected using /opt/sbin/kats/cannot-exec.real.
- Kernel, libmd, OpenSSL, QuickSec, SSH Ipsec – verifies correct output from known answer tests for appropriate algorithms.

121. 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 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.

122. 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 [ECG].

123. 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

124. 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/ND, FMT_MOF.1/ManualUpdate,***)

125. The installable firmware package includes the full Junos OS firmware. No partial updates are supported. 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.

126. 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.

127. The fingerprint loader will only process a manifest for which it can verify the signature. 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.

128. 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. Passwords used for authentication can be viewed by Security Administrators using the CLI command ‘request system decrypt password’ (***FIA_X.509_EXT.1/Rev, FMT_MTD.1/CoreData***).

129. 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

130. Junos OS creates and stores audit records for the following events (the detail of content recorded for each audit event is detailed in Table 5 (***FAU_GEN.1/ND***) and Table 6 (***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 Administrator actions for configuring two instances of the TOE in Cluster Mode
- All state synchronisations between two instances of the TOE in Cluster Mode
- 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
- Unsuccessful attempt to validate a certificate.
- Any addition, replacement or removal of trust anchors in the TOE's trust store.
- 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

131. 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.
132. 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.
133. 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).
134. 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.
135. Suppressed logs are reported as single log entries containing the count of occurrences

136. 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 ***created*** 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

137. 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

138. 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

139. 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).

140. 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" action is performed and the whole appliance is zeroized (which by definition cannot be recorded)

141. 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, which is maintained using the hardware Time Stamp Counter as the clock source. (***FAU_GEN.2, FPT_STM_EXT.1***)

142. 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 (***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.

143. 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.

144. 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.

7.6 Management

145. 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***)

146. 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/ND***)

147. 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:
    - Transmission of audit data to an external IT entity, 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/ND, FMT_SMF.1/IPS***)
    - Configuring the packet filtering tules of the TOE (***FMT_SMF.1/FFW***)
    - Handling of audit data, 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/IPS, FMT_SMF.1/ND, FMT_SMF.1/FFW***):
   - 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[39], with default value of 180 seconds), IPsec lifetime-seconds (within range 180 to 86400, with default value of 28800 seconds[40]), 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 [MOD_VPNGW] (***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***);

[39] 180 to 86400 seconds is a range of 3 minutes to 24 hours.
[40] 28800 seconds is 8 hours.

- 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***).

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

7.7 User Data Protection

149. 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 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

150. 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

151. 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.

152. 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.

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

154. 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.

155. 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. 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`'.

156. 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

157. 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.

158. 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.

159. 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.

160. 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.

161. 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.

162. 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.

163. 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.

164. 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.

165. 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 12 Traffic Filtering RFCs

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

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

168. 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

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

170. 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**)

171. 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.

172. 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

173. 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.

174. For more information about configuring event logging, refer to [ECG] (***FFW_RUL_EXT.1, FPF_RUL_EXT.1***)

7.9 Intrusion Prevention System

175. 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.

176. 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[41].

177. 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***)

178. 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.

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

- 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
    - 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)

179. 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.

180. 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***)

181. 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***)

182. 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***)

183. 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.

184. 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***)

185. 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***)

186. 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	n/a (configured by default)
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 13 IPS signature names

187. 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”.

188. 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”[42];

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

- 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***)

189. 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.

190. 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).

191. 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 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***)

192. For more information about configuring event logging, refer to [ECG]. (***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
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	FIA_X509_EXT.2	FIA_X509_EXT.2 included
FIA_X509_EXT.2	FIA_X509_EXT.1	FIA_X509_EXT.1 included
FIA_X509_EXT.3	FCS_CKM.1 Cryptographic Key Generation FIA_X509_EXT.1	FCS_CKM.1/ND included FIA_X509_EXT.1 included
FIA_PSK_EXT.1	FCS_COP.1 FCS_IPSEC_EXT.1 FCS_RBG_EXT.1	FCS_COP.1 included FCS_IPSEC_EXT.1 included FCS_RBG_EXT.1 included
FMT_MOF.1/ManualUpdate	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1/ND included
FMT_MOF.1/Services	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1/ND, FMT_SMF.1/IPS, FMT_SMF.1/FFW included
FMT_MOF.1/Functions	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1/ND included
FMT_MTD.1/CoreData	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1/ND included
FMT_MTD.1/CryptoKeys	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1/ND included
FMT_SMF.1/ND	None	n/a
FMT_SMF.1/IPS	None	n/a
FMT_SMF.1/FFW	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

Security Functional Requirement	Dependency	Rationale
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
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	FFW_RUL_EXT.1	FFW_RUL_EXT.1 included
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 14 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
CFP	C Form-factor Pluggable
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
GPIM	Gigabit-Backplane PIM
HA	High Availability
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
IOC	I/O (Input/Output) Cards
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
JET	Junos Extension toolkit. Control plane APIs for Junos.
KAS	Key Agreement Scheme
Junos	Juniper Operating System
LDP	Label Distribution Protocol
MAC	Mandatory Access Control
MIC	Modular Interface Cards
MPC	Modular Port Concentrator
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
POE	Power over Ethernet
PP	Protection Profile
PRNG	Pseudo Random Number Generator
QSFP	Quad SFP
RE	Routing Engine
RFC	Request for Comment
RIP	Routing Information Protocol
RNG	Random Number Generator
RSA	Rivest, Shamir, Adelman
SA	Security Association
SCB	Switch Control Board
SCEP	Simple Certificate Enrollment Protocol
SFP	Small Form-factor Pluggable
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SNMP	Simple Network Management Protocol
SPC	Services Processing Card
SPU	Services Processing Units
SSC	Shared Secret Computation
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
VPN	Virtual Private Network
VPNEP	Virtual Private Network Extended Package
VDSL	Very-high-bit-rate Digital Subscriber Line
XFP	10 Gigabit SFP