Security Target Junos OS 18.1R2 for QFX10002, QFX10008 and QFX100016

Juniper Networks

Version 1.5

January 2019

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

Abstract

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), Junos OS 18.1R2 for QFX10K Series. This Security Target (ST) is conformant to the requirements of Collaborative Protection Profile for Network Devices ([NDcPP]) v2.0E.

References

[CC1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, CCMB-2012-09-001, Version 3.1 Revision 4, September 2012

[CC2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, CCMB-2012-09-002, Version 3.1 Revision 4, September 2012

[CC3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, CCMB-2012-09-003, Version 3.1 Revision 4, September 2012

[CEM] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, CCMB-2012-09-004, Version 3.1 Revision 4, September 2012

[CC_Add] CC and CEM Addenda, Exact Conformance, Selection-Based SFRs, Optional SFRs, CCDB-2017-05-xxx, Version 0.5, May 2017

[NDcPP] Collaborative Protection Profile for Network Devices, version 2.0+Errata 20180314, dated 14 March 2018

[SD] Supporting Document, Evaluation Activities for Network Device cPP, March 2018, version 2.0+Errata 20180314

Table of Contents

1 Introduction

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

1.1 ST reference

ST Title	Security Target Junos OS 18.1R2 for QFX10002, QFX10008 and QFX100016
ST Revision	1.5
ST Draft Date	January 2019
Author	Juniper Networks, Inc.
cPP/EP Conformance	Collaborative Protection Profile for Network Devices, version 2.0+Errata 20180314, dated 14 March 2018

1.2 TOE Reference

TOE Title	Junos OS 18.1R2 for QFX10K Series
TOE Firmware	Junos OS 18.1R2-S3

1.3 About this document

2. This Security Target follows the following format:

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

Table 1 Document Organization

1.4 Document Conventions

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

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

  e.g. "[*selection: disclosure, modification, loss of use*]" in [CC2] or an ECD might become "<u>disclosure</u>" (completion;

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

  e.g. "[selection: *change_default, query, modify, delete, [assignment: other operations*]]" in [CC2] or an ECD might become "change_default, *select_tag*" (completion of both selection and assignment);
- Iteration: indicated by adding a string starting with "/" (e.g. "FCS_COP.1/Hash").

4. Operations performed in [NDcPP] are not marked in this Security Target.

1.5 TOE Overview

5. The Target of Evaluation (TOE) is Juniper Networks, Inc. Junos OS 18.1R2 for QFX10K Series executing on QFX10K-Series Ethernet Switches. The supported QFX10K-Series chassis are:
   - QFX10002
   - QFX10008
   - QFX10016
6. Each of the Ethernet Switches is a secure network device that protects itself largely by offering only a minimal logical interface to the network and attached nodes. All switching platforms are powered by the Junos OS firmware, Junos OS 18.1R2 for QFX10K Series, which is a special purpose OS that provides no general purpose computing capability. Junos OS provides both management and control functions as well as all IP switching.
7. The Ethernet Switches primarily support the definition of, and enforce, information flow policies among network nodes. All information flow from one network node to another passes through an instance of the TOE. Information flow is controlled on the basis of network node addresses and protocol. In support of the information flow security functions, the TOE ensures that security-relevant activity is audited, and provides the security tools to manage all of the security functions.

1.6 TOE Description

1.6.1 Overview

8. Each Juniper Networks QFX10K Ethernet Switch delivers industry-leading scalability, density, and flexibility, helping cloud and data center operators build automated data center networks that provide superior long-term investment protection. Designed for a diverse set of deployment options, the QFX10K switches allow data center operators to build cloud networks that best suit their deployment needs and easily evolve as requirements change over time.
9. The appliances are physically self-contained, housing the firmware and hardware necessary to perform all switching functions.
   - The QFX10002 Ethernet Switches are a fixed chassis configuration switch, offering 72-port (QFX10002-72Q) and 36-port (QFX10002-36Q) 40GbE options (supporting quad small form-factor pluggable plus transceiver (QSFP+) and QSFP28 ports).
   - The QFX10008 and QFX10016 Ethernet Switches are modular appliances, comprised of the hardware components:
     - The Modular Ethernet switch chassis, populated with Control Boards (one or more instances of the QFX10000 Control Board)

- Line cards installed in the chassis, which allow the appliance to communicate with the different types of networks that may be required within the environment where the Ethernet Switches are used.

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

- The Routing Engine (RE) runs the Junos firmware and provides Layer 2 and Layer 3 switching services and network management for all operations necessary for the configuration and operation of the TOE and controls the flow of information through the TOE. The RE is fixed in the QFX10002 Ethernet Switch, while the RE is the pluggable QFX10000 Control Board in the QFX10008 and QFX10016 chassis.
- The Packet Forwarding Engine (PFE) provides all operations necessary for 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 Ethernet Switches support numerous switching standards for flexibility and scalability.

13. The functions of the Ethernet Switches can all be managed through the Junos firmware, either from a connected terminal console or via a network connection. Network management is secured using the SSH protocol. All management, whether from a user connecting to a terminal or from the network, requires successful authentication. In the evaluated deployment the TOE is managed and configured via Command Line Interface, either via a directly connected console or over the network secured using the SSH protocol.

14. The Junos RE functionality is running as a Guest in a Virtual Machine (VM) provided by Wind River Linux (WRL7). The WRL virtualisation is provided using an optimized Kernel-Based Virtual Machine (KVM).

1.6.2 Physical boundary

15. The TOE is the Junos OS 18.1R2 for QFX10K Series firmware running on the appliance chasses listed in Table 2. Hence the **TOE is contained within the physical boundary of the specified appliance chassis** as shown in Figure 1 below.

16. The **physical boundary of the TOE is:**

- QFX10002 Ethernet Switches – the entirety of the Fixed Ethernet Switch appliance (either QFX10002-72Q or QFX10002-36Q switch)
- QFX10008 and QFX10016 Modular Ethernet Switches – the chassis populated with at least one instance of the QFX10000 Control Board and one or more or the line cards listed in Table 2.

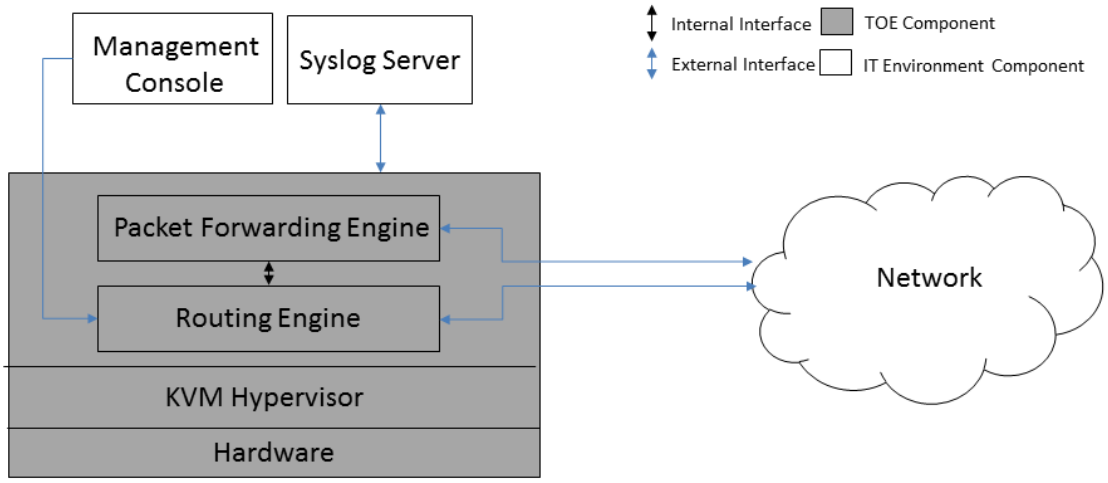

Figure 1 TOE Boundary

17. Separate jinstall images are provided for QFX10002 and QFX10008/QFX10016, namely:

- **QFX10002**: jinstall-host-qfx-10-f-x86-64-18.1R2-S3.3-secure-signed.tgz
- **QFX10008/QFX10016**: jinstall-host-qfx-10-m-x86-64-18.1R2-S3.3-secure-signed.tgz

18. The TOE interfaces comprise the following:

i. Network interfaces which pass traffic

ii. Management interface through which handle administrative actions.

Ethernet Switch Model	Network Ports	Routing Engine	Operating System and Processor
QFX10002	QSFP+ for 40GbE speeds QSFP28 ports for 100GbE speeds	Fixed in QFX10002 chassis	Junos OS 18.1R2 Intel Xeon E3
QFX10008	• QFX10000-36Q, a 36-port 40GbE quad small form-factor pluggable plus transceiver (QSFP+) or 12-port 100GbE QSFP28 line card • QFX10000-30C, a 30-port 100GbE QSFP28/40GbE QSFP+ line card • QFX10000-60S-6Q, a 60-port 1GbE/10GbE SFP/SFP+ line card with six-port 40GbE QSFP+ / two-port 100GbE QSFP28 • QFX10008 Switch Fabric	QFX10000 Control Board	

Ethernet Switch Model	Network Ports	Routing Engine	Operating System and Processor
QFX10016	• QFX10000-36Q, a 36-port 40GbE quad small form-factor pluggable plus transceiver (QSFP+) or 12-port 100GbE QSFP28 line card • QFX10000-30C, a 30-port 100GbE QSFP28/40GbE QSFP+ line card • QFX10000-60S-6Q, a 60-port 1GbE/10GbE SFP/SFP+ line card with six-port 40GbE QSFP+ / two-port 100GbE QSFP28 • QFX10016 Switch Fabric	QFX10000 Control Board	

Table 2 TOE Chassis Details

19. The firmware version reflects the detail reported for the components of the Junos OS when the "show version" command is executed on the appliance.

20. The guidance documents included as part of the TOE are:

[ECG] Junos OS Common Criteria and FIPS Evaluated Configuration Guide for QFX10002, QFX10008, and QFX10016 Series Devices, Release 18.1R2.

1.6.3 Logical Boundary

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

Security Functionality	Description
Protected Communications	The TOE provides an SSH server to support protected communications for administrators to establish secure sessions and to connect to external syslog servers. The TOE requires that applications exchanging information with it are successfully authenticated prior to any exchange (i.e. applications connecting over SSH). The TOE includes cryptographic modules that provide the underlying cryptographic services, including key management and protection of stored keys, algorithms, random bit generation and crypto-administration. The cryptographic modules provide confidentiality and integrity services for authentication and for protecting communications with connecting applications.
Administrator Authentication	Administrative users must provide unique identification and authentication data before any administrative access to the system is granted. Authentication data entered and stored on the TOE is protected. The TOE can be configured to terminate interactive user sessions and to present an access banner with warning messages prior to authentication.
Correct Operation	The TOE provides for both cryptographic and non-cryptographic self-tests, and is capable of automated recovery from failure states.
Trusted Update	The administrator can initiate update of the TOE firmware. The integrity of any firmware updates is verified prior to installation of the updated firmware.

Security Functionality	Description
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, as well as the events listed in Table 4. Audit records include the date and time, event category, event type, username, and the outcome of the event (success or failure). Local syslog storage limits are configurable and are monitored. In the event of storage limits being reached the oldest logs will be overwritten.
Management	The TOE provides a Security Administrator role that is responsible for: • the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the evaluated product • the regular review of all audit data; • initiation of trusted update function; • all administrative tasks (e.g., creating the security policy). The devices are managed through a Command Line Interface (CLI). The CLI is accessible through local (serial) console connection or remote administrative (SSH) session.

1.6.4 Non-TOE hardware/software/firmware

22. The QFX10K-series models of the TOE require network interface (SFP/SFP+ ports in the case of the QFX10002 and line cards for QFX10008/QFX10016as detailed in Table 2) to operate and communicate with the connected network.

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

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

1.6.5 Summary of out scope items

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

2 Conformance Claim

2.1 CC Conformance Claim

24. This Security Target conforms to the requirements of Common Criteria v3.1, Revision 4 and is Part 2 extended and Part 3 conformant.

2.2 PP Conformance claim

25. This Security Target claims Exact Conformance to:

    - Collaborative Protection Profile for Network Devices, version 2.0+Errata 20180314, dated 14 March 2018 [NDcPP]

26. Exact conformance is defined in [NDcPP] section 2 and in [CC_Add].

27. The Security Problem definition in this Security Target is consistent with the security problem definitions detailed in the collaborative Protection Profile to which this ST claims conformance, namely:

    - [NDcPP] Section 4.

28. The statement of the Security Problem Definition in this ST is identical to the set of the threats, organizational security policies and assumptions from the collaborative Protection Profile. Hence, this SPD statement is considered to be conformant to the collaborative Protection Profiles and Extended Packages claimed.

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

    - The prose in [NDcPP] Section 3.

30. Again, the statement of the Security Objectives in this ST is identical to the set of the security objectives from the collaborative Protection Profile. Hence, the statement of security objectives in this ST is considered to be conformant to the collaborative Protection Profile claimed.

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

    - [NDcPP] Sections 6 & 7.

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

33. This statement of SFRs is augmented with SFRs specified in [NDcPP] Appendix A (Optional requirements).

34. All extended requirements in this ST are taken from [NDcPP] Appendix C.

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

    - [NDcPP] Section 7.

36. Hence, the statement of security requirements in this ST is considered to be conformant to the collaborative Protection Profile claimed.

37. 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

38. In line with Labgram #105, this section identifies all NIAP Technical Decisions that are applicable to [NDcPP] and states whether each is applicable to this TOE:

TD	REFERENCE	Applicable	Exclusion Rationale
TD0343: NIT Technical Decision for Updating FCS_IPSEC_EXT.1.14 Tests	ND SD V2.0, FCS_IPSEC_EXT.1.14	No	This TD is associated with FCS_IPSEC_EXT.1. The TOE does not include FCS_IPSEC_EXT.1 functionality.
TD0342: NIT Technical Decision for TLS and DTLS Server Tests	ND SD V2.0, FCS_DTLSS_EXT.1, FCS_DTLSS_EXT.2, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2	No	This TD is associated with FCS_[D]TLSS_EXT.x. The TOE does not include FCS_[D]TLSS_EXT.x functionality.
TD0341: NIT Technical Decision for TLS wildcard checking	ND SD V2.0, FCS_TLSC_EXT.1.2, FCS_TLSC_EXT.2.2, FCS_DTLSC_EXT.1.2, FCS_DTLSC_EXT.2.2	No	This TD is associated with FCS_[D]TLSC_EXT.x. The TOE does not include FCS_[D]TLSC_EXT.x functionality.
TD0340: NIT Technical Decision for Handling of the basicConstraints extension in CA and leaf certificates	FIA_X509_EXT.1.1	Yes	
TD0339: NIT Technical Decision for Making password-based authentication optional in FCS_SSHS_EXT.1.2	ND SD V2.0, FCS_SSHS_EXT.1.2	Yes	
TD0338: NIT Technical Decision for Access Banner Verification	ND SD V2.0, FTA_TAB.1	Yes	
TD0337: NIT Technical Decision for Selections in FCS_SSH*_EXT.1.6	ND SD V2.0, FCS_SSHC_EXT.1, FCS_SSHS_EXT.1	Yes	
TD0336: NIT Technical Decision for Audit requirements for FCS_SSH*_EXT.1.8	ND SD V2.0, FCS_SSHC_EXT.1.8, FCS_SSHS_EXT.1.8	Yes	
TD0335: NIT Technical Decision for FCS_DTLS Mandatory Cipher Suites	FCS_DTLSC_EXT.1.1, FCS_DTLSC_EXT.2.1, FCS_DTLSS_EXT.1.1, FCS_DTLSS_EXT.2.1, FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.2.1, FCS_TLSS_EXT.1.1, FCS_TLSS_EXT.2.1	No	This TD is associated with FCS_[D]TLS*_EXT.x. The TOE does not include FCS_[D]TLS*_EXT.x functionality.
TD0334: NIT Technical Decision for Testing SSH when password-based authentication is not supported	ND SD V2.0, FCS_SSHC_EXT.1.9	No	This TD is associated with FCS_SSHC_EXT.1. The TOE does not include FCS_SSHC_EXT.1 functionality.

TD	REFERENCE	Applicable	Exclusion Rationale
TD0333: NIT Technical Decision for Applicability of FIA_X509_EXT.3	ND SD V2.0, FIA_X509_EXT	Yes	
TD0324: NIT Technical Decision for Correction of section numbers in SD Table 1	Table 1, CPP_ND_V2.0E	Yes	
TD0323: NIT Technical Decision for DTLS server testing - Empty Certificate Authorities list	ND SD V2.0, FCS_DTLSS_EXT.2.7, FCS_DTLSS_EXT.2.8, CPP_ND_V2.0E	No	This TD is associated with FCS_DTLSS_EXT.2. The TOE does not include FCS_DTLSS_EXT.2 functionality.
TD0322: NIT Technical Decision for TLS server testing - Empty Certificate Authorities list	ND SD V.1.0, ND SD V2.0, FCS_TLSS_EXT.2.4, FCS_TLSS_EXT.2.5, CPP_ND_V2.0E	No	This TD is associated with FCS_TLSS_EXT.2. The TOE does not include FCS_TLSS_EXT.2 functionality.
TD0321: Protection of NTP communications	FTP_ITC.1, FPT_STM_EXT.1, CPP_FW_V2.0E, CPP_ND_V2.0E	No	The evaluation does not include setting time remotely. This TD addresses NTP communications.
TD0291: NIT Technical Decision for DH14 and FCS_CKM.1	FCS_CKM.1 CPP_FW_V1.0, CPP_FW_v2.0, CPP_FW_V2.0E, CPP_ND_V1.0, CPP_ND_V2.0, CPP_ND_V2.0E, ND SD V.1.0, ND SD V2.0	Yes	
TD0290: NIT Technical Decision for physical interruption of trusted path/channel	FTP_ITC.1, FTP_TRP.1, FPT_ITT.1 CPP_ND_V1.0, CPP_ND_V2.0, CPP_ND_V2.0E, ND SD V.1.0, ND SD V2.0	Yes	
TD0289: NIT Technical Decision for FCS_TLSC_EXT.x.1 Test 5e	CPP_ND_V1.0, CPP_ND_V2.0, CPP_ND_V2.0E FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.2.1, FCS_DTLSC_EXT.1.1 (only ND SD V2.0) , FCS_DTLSC_EXT.2.1 (only ND SD V2.0)	No	This TD is associated with FCS_[D]TLSS_EXT.x. The TOE does not include FCS_[D]TLSS_EXT.x functionality.
TD0281 : NIT Technical Decision for Testing both thresholds for SSH rekey	FCS_SSHC_EXT.1.8, FCS_SSHS_EXT.1.8, CPP_ND_V1.0, CPP_ND_V2.0, CPP_ND_V2.0E ND SD V1.0, ND SD V2.0	Yes	

TD	REFERENCE	Applicable	Exclusion Rationale
TD0262 : NIT Technical Decision for TLS server testing - Empty Certificate Authorities list	CPP_ND_V1.0, CPP_ND_V2.0 ND SD V.1.0, ND SD V2.0, FCS_TLSS_EXT.2.4, FCS_TLSS_EXT.2.5	No	This TD has been archived.
TD0260: NIT Technical Decision for Typo in FCS_SSHS_EXT.1.4	CPP_FW_v2.0, CPP_FW_V2.0E, CPP_ND_V2.0, CPP_ND_V2.0E FCS_SSHS_EXT.1.4	No	This TD has been archived.
TD0259: NIT Technical Decision for Support for X509 ssh rsa authentication IAW RFC 6187	CPP_FW_v2.0, CPP_FW_V2.0E, CPP_ND_V2.0, CPP_ND_V2.0E FCS_SSHC_EXT.1.5/FCS_SSHS_EXT.1.5	Yes	
TD0257: NIT Technical Decision for Updating FCS_DTLSC_EXT.x.2/FCS_TLSC_EXT.x.2 Tests 1-4	ND SD V1.0, ND SD V2.0, FCS_DTLSC_EXT.1.2/FCS_DTLSC_EXT.2.2 Tests 1-4 (ND SD V2.0), FCS_TLSC_EXT.1.2/FCS_TLSC_EXT.2.2, Tests 1-4 (ND SD V1.0, ND SD V2.0) CPP_ND_V1.0, CPP_ND_V2.0, CPP_ND_V2.0E	No	This TD is associated with FCS_[D]TLSS_EXT.2. The TOE does not include FCS_[D]TLSS_EXT.2 functionality.
TD0256: NIT Technical Decision for Handling of TLS connections with and without mutual authentication	ND SD V1.0, ND SD V2.0, FCS_DTLSC_EXT.2.5 (ND SD V2.0), FCS_TLSC_EXT.2 (ND SD V1.0, ND SD V2.0) CPP_ND_V1.0, CPP_ND_V2.0, CPP_ND_V2.0E	No	This TD is associated with FCS_[D]TLSC_EXT.2. The TOE does not include FCS_[D]TLSC_EXT.2 functionality.
TD0228: NIT Technical Decision for CA certificates - basicConstraints validation	ND SD V1.0, ND SD V2.0, FIA_X509_EXT.1.2 CPP_FW_V1.0, CPP_ND_V1.0, CPP_ND_V2.0, CPP_ND_V2.0E	Yes	

Table 3 Applicable NIAP Technical Decisions

39. 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

- This TD relates to a configuration not supported by this TOE
- Relates to (selection-based) claims not included in the scope of this TOE

3 Security Problem Definition

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

3.1 Threats

41. The following threats for this TOE are as defined in [NDcPP] Section 4.1, namely:

- T.UNAUTHORIZED_ADMINISTRATOR_ACCESS
- T.WEAK_CRYPTOGRAPHY
- T.UNTRUSTED_COMMUNICATION_CHANNELS
- T.WEAK_AUTHENTICATION_ENDPOINTS
- T.UPDATE_COMPROMISE
- T.UNDETECTED_ACTIVITY
- T.SECURITY_FUNCTIONALITY_COMPROMISE
- T.PASSWORD_CRACKING
- T.SECURITY_FUNCTIONALITY_FAILURE

42. No threats are identified for this TOE in addition to those specified in the collaborative Protection Profile.

3.2 Assumptions

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

- A.PHYSICAL_PROTECTION
- A.LIMITED_FUNCTIONALITY
- A.NO_THRU_TRAFFIC_PROTECTION
- A.TRUSTED_ADMINSTRATOR
- A.REGULAR_UPDATES
- A.ADMIN_CREDENTIALS_SECURE
- A.RESIDUAL_INFORMATION

44. No assumptions are identified for this TOE in addition to those specified in the collaborative Protection Profiles.

3.3 Organizational Security Policies

45. The OSPs applied for this TOE are as defined in [NDcPP] Section 4.3, namely:

- P.ACCESS_BANNER

No additional OSPs are identified and no modification to the statement of OSPs is made for this TOE.

4 Security Objectives

46. 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

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

4.2 Security Objectives for the Operational Environment

48. The statement of security objectives for the operational environment of this TOE is as defined in [NDcPP] Section 5.1, namely:

- OE.PHYSICAL
- OE.NO_GENERAL_PURPOSE
- OE.NO_THRU_TRAFFIC_PROTECTION
- OE.TRUSTED_ADMIN
- OE.UPDATES
- OE.ADMIN_CREDENTIALS_SECURE
- OE.RESIDUAL_INFORMATION

4.3 Security Objectives rationale

49. As these objectives for the TOE and operational environment are the same as those specified in [NDcPP],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 to which this ST claims conformance:

- [NDcPP] section 4.

5 Security Functional Requirements

50. All security functional requirements are taken from the [NDcPP]. The SFRs are presented in accordance with the conventions described in [NDcPP] Section 6.1, and section 1.4 of this document.

51. 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 Audit Data Generation

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

a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified 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).*
- *[Starting and stopping services];*

d) Specifically defined auditable events listed in Table 4.

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

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

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the ST, *information specified in column three of Table 4*.

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG_EXT.1	None	None
FAU_STG.1	None	None
FCS_CKM.1	None	None
FCS_CKM.2	None	None
FCS_CKM.4	None	None
FCS_COP.1/DataEncryption	None	None
FCS_COP.1/SigGen	None	None

FCS_COP.1/Hash	None	None
FCS_COP.1/KeyedHash	None	None
FCS_RBG_EXT.1	None	None
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.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None	None
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None
FMT_MTD.1/CoreData	All management activities of TSF data	None
FMT_SMF.1	None	None
FMT_SMR.2	None	None
FPT_SKP_EXT.1	None	None
FPT_APW_EXT.1	None	None
FPT_TST_EXT.1	None	None
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	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 (if “terminate the session” is selected)	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	Reason for failure
FIA_X509_EXT.2	None	None
FPT_TUD_EXT.2	Failure of update	Reason for failure (including identifier of invalid certificate)
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.

Table 4 FAU_GEN.1 Security Functional Requirements and Auditable Events

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 data 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.

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.

5.1.2.2 FAU_STG.1 Protected audit trail storage (Optional)

FAU_STG.1 Protected audit trail storage

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

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

5.2 Cryptographic Support (FCS)

5.2.1 Cryptographic Key Management (FCS_CKM)

5.2.1.1 FCS_CKM.1 Cryptographic Key Generation (Refinement)

FCS_CKM.1 Cryptographic Key Generation

FCS_CKM.1.1 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

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

5.2.1.2 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*]~~.

5.2.1.3 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)

FCS_COP.1.1/DataEncryption The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [CBC, CTR] *mode* and cryptographic key sizes *[128 bits,*

256 bits] that meet the following: *AES as specified in ISO 18033-3, [CBC as specified in ISO 10116, CTR as specified in ISO 10116]*.

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, 3072 bits],*
- *Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits]*

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

that meet the following: *[*

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

].

FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

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

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

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm *[HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512]* and cryptographic key sizes *[160, 256 and 512 bits]* **and message digest sizes [160, 256, 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 [*[5]* software-based noise source, *[1-2[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.

[1] The QFX10002 has a single hardware noise source; the QFX10008 and QFX10016 have 2 hardware noise sources.

5.2.4 Cryptographic Protocols (Extended – FCS_SSHS_EXT SSH Protocol)

5.2.4.1 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1 SSH Server Protocol

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

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 [ecdsa-sha2-nistp256] 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 data integrity 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 no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

5.3 Identification and Authentication (FIA)

5.3.1 Authentication Failure Management (FIA_AFL)

5.3.1.1 FIA_AFL.1 Authentication Failure Management (Refinement)

FIA_AFL.1 Authentication Failure Management

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*.

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

- *prevent the offending remote Administrator from successfully authenticating 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, 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

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

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.
- 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.*
- *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 [no protocols], and [code signing for system software updates].

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [accept the certificate].

ST Application Note:

The first selection operation in FIA_X509_EXT.2.1 is completed with "none" in accordance with the [NDcPP] Application Note 121, which states

> "The ST author must include FIA_X509_EXT.2 in all instances except when only SSH is selected within FTP_ITC.1 or FPT_ITT.1 and ssh-rsa, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, and/or ecdsa-sha2-nistp521 authentication is also selected."

5.4 Security Management (FMT)

5.4.1 Management of functions in TSF (FMT_MOF)

5.4.1.1 FMT_MOF.1/ManualUpdate Management of security functions behaviour

FMT_MOF.1/ManualUpdate Management of security functions behaviour

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

5.4.1.2 FMT_MOF.1/Services Management of security functions behaviour

FMT_MOF.1/Services Management of security functions behaviour

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

5.4.1.3 FMT_MOF.1/Functions Management of security functions behaviour

FMT_MOF.1/Functions Management of security functions behaviour

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

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

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to *manage* the *cryptographic keys* 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 Specification of Management Functions

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

[

- *Ability to administer the TOE locally and remotely;*
- *Ability to 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] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- *[*
  - *Ability to configure audit behaviour;*
  - *Ability to configure the cryptographic functionality;*
  - *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]*

]

5.4.4 Security management roles (FMT_SMR)

5.4.4.1 FMT_SMR.2 Restrictions on security roles

FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator*.

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

5.5 Protection of the TSF (FPT)

5.5.1 Protection of TSF Data (Extended – FPT_SKP_EXT)

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

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

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

5.5.2 Protection of Administrator Passwords (Extended – FPT_APW_EXT)

5.5.2.1 FPT_APW_EXT.1 Protection of Administrator Passwords

FPT_APW_EXT.1 Protection of Administrator Passwords

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

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

5.5.3 TSF testing (Extended – FPT_TST_EXT)

5.5.3.1 FPT_TST_EXT.1 TSF Testing (Extended)

FPT_TST_EXT.1 TSF testing

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

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

5.5.4 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] prior to installing those updates.

5.5.4.2 FPT_TUD_EXT.2 Trusted Update based on certificates

FPT_TUD_EXT.2 Trusted Update based on certificates

FPT_TUD_EXT.2.1 The TSF shall not install an update if the code signing certificate is deemed invalid.

FPT_TUD_EXT.2.2 When the certificate is deemed invalid because the certificate has expired, the TSF shall [not accept the certificate].

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].

5.6 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*.

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 **be capable of using [SSH] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [no other capabilities]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **disclosure and detection of modification of the channel data**.

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

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [*no communication*].

5.7.2 Trusted Path (FTP_TRP)

5.7.2.1 FTP_TRP.1/Admin Trusted Path (Refinement)

FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin The TSF shall **be capable of using [SSH] to** provide a communication path between itself and **authorized** remote **Administrators** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from **disclosure and provides detection of modification of the channel data**.

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

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

6 Security Assurance Requirements

52. The TOE security assurance requirements are taken from [NDcPP] , together with the refinements documented in [NDcPP] Section 7, as listed in Table 5 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)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.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 5 Security Assurance Requirements

7 TOE Summary Specification

7.1 Protected communications

53. 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

54. All FIPS-approved cryptographic functions implemented by the switch are implemented in the following modules:

- OpenSSL
- LibMD
- Kernel

55. All random number generation by the TOE is performed in accordance with NIST Special Publication 800-90 using HMAC_DRBG implemented in the OpenSSL module and Kernel module (FCS_RBG_EXT.1.1). Additionally, SHA (256,512) is implemented in the LibMD module which is used for password hashing by Junos' MGD daemon. **The switch is to be operated with FIPS mode enabled.**

56. 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 6.

Implementation	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	Certificate Number
OpenSSL (OpenSSH)	FIPS 197, SP 800-38A	AES-CBC (128, 256) (Encrypt, Decrypt)	5459
	FIPS 180-4	SHA1, SHA2-256, SHA2-384, SHA2-512 (byte Oriented) (Message Digest Generation)	4381
	FIPS 198-1	HMAC-SHA1, HMAC-SHA2-256, HMAC-SHA2-512 (byte Oriented) (Message Authentication)	3617
	FIPS 186-4	ECDSA (P-256 w/ SHA-256) ECDSA (P-384 w/ SHA-384) ECDSA (P-521 w/ SHA-521) (SigGen, SigVer, KeyGen, KeyVer for ECDH)	1458
	FIPS 186-4	RSA PKCS1_V1_5[2] (n=2048 (SHA 256), n=3072 (SHA 256)) (SigGen, SigVer, KeyGen)	2931
	SP 800-56A	CVL/KAS ECC Key Agreement Public key Validation, Key Pair Generation Initiator, Responder EC (P-256, SHA-256), ED (P-384, SHA-384), EE (P-521, SHA-512)	1908
OpenSSL	SP 800-90A	DRBG (HMAC-SHA-2-256) Prediction Resistance: Enabled (Random Bit Generation)	2142

[2] Including PKCS#1 v1.5 padding

Implementation	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	Certificate Number
LibMD	FIPS 180-4	SHA1, SHA2-256, SHA2-512 (byte Oriented) (Message Digest Generation)	4380
	FIPS 198-1	HMAC-SHA1, HMAC-SHA2-256 (byte Oriented)	3616
Kernel	FIPS 197, SP 800-38A	AES-CBC (128, 256) (Encrypt, Decrypt)	5458
	FIPS 180-4	SHA1, SHA2-256, SHA2-384, SHA2-512 (byte Oriented) (Message Digest Generation)	4379
	FIPS 198-1	HMAC-SHA1, HMAC-SHA2-256 (byte Oriented) (Message Authentication)	3615
	SP 800-90A	DRBG (HMAC-SHA-2-256) Prediction Resistance: Enabled (Random Bit Generation)	2141

Table 6 CAVP Certificate Results for Cryptographic Services

57. The FIPS approved algorithms are applied when the FIPS mode is enabled[3]. The relevant FIPS knobs are specified in [ECG]. (***FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, FCS_COP.1/KeyedHash, FCS_RBG_EXT.1, FCS_CKM.1, FMT_SMF.1***)

58. Asymmetric keys are 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 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. The TOE implements Diffie-Hellman group 14, using the modulus and generator specified by Section 3 of RFC3526. (***FCS_CKM.2, FCS_CKM.1***)

59. The following table relates cryptographic algorithms to the protocols by the TOE. The TOE acts as the server for SSH as listed in Table 7:

Protocol	Key Exchange	Auth	Cipher	Integrity
SSHv2	ECDH-sha2-nistp256 ECDH-sha2-nistp384 ECDH-sha2-nistp521 Diffie-Hellman group 14(modp 2048)	ECDSA P-256	AES CTR 128 AES CTR 256 AES CBC 128 AES CBC 256	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512

Table 7 Supported Protocols

60. Junos OS handles zeroization for all CSP, plaintext secret and private cryptographic keys according to Table 8 below. (***FCS_CKM.4***).

[3] The knob “set system fips level 1” will enforce strict compliance to FIPS and enable restrictions on algorithms and keys sizes as required by FIPS requirements.

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)	Plaintext	File format on Virtual disk (mapped to SDD)/Memory	When the appliance is recommissioned, the config files (including CSP files such as SSH keys) are removed using the Linux `rm-f` command to wipe the underlying persistent storage media. The `request system zeroize` option should be used during recommissioning.
	Loaded into memory to complete session establishment	Plaintext	Memory	Memory `free()` operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)
SSH Session Key	Session keys used with SSH, AES 128, 256, hmac-sha-1, hmac-sha2-256 or hmac-sha2-512 key (160, 256 or 512), DH Private Key (2048 or elliptic curve 256/384/521-bits)	Plaintext	Memory	Memory `free()` operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)
User Password	Plaintext value as entered by user	Plaintext as entered	Processed in Memory	Memory `free()` operation is performed by Junos upon completion of authentication (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)
		Hashed when stored (HMAC-sha1, sha256, sha512)	Stored on Virtual disk (mapped to SDD)/Memory	When the appliance is recommissioned, the config files (including the obfuscated password) are removed using the Linux `rm-f` command command to wipe the underlying persistent storage media. The `request system zeroize` option.
DRBG State	Internal state and seed key of DRBG	Plaintext	Memory	Handled by Kernel, overwritten with zero's at reboot.
ecdh private keys	Loaded into memory to complete key exchange in session establishment	Plaintext	Memory	Memory `free()` operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)

Table 8 CSP Storage and Zeroization

61. 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[4]. (***FPT_SKP_EXT.1***)

7.1.2 Random Bit Generation

62. 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 QFX10K Series platforms rely on multiple noise sources for the RBG:

- RANDOM_NET_ETHER: This software entropy source contributes entropy on interrupt from network DMA. The cycle count along with 256 bits from an internal packet data buffer is collected. The packet data is squashed to a 4 bytes hash using a Jenkins hash algorithm plus 64 bits of cycle counter. This entropy is fed into HMAC_DRBG.
- RANDOM_PURE: 16 octets are taken from the hardware noise source(s). In the QFX10002, this is a single source - the processor RDRAND output. For the QFX10008 and QFX10016, there are 2 hardware sources - RDRAND is used along with output from the TPM.
- RANDOM_NET: This software source of entropy is associated with network activity that is destined for the Junos OS. The transit network traffic is not involved in this process.
- RANDOM_INTERRUPT: This software source of entropy is associated with hardware devices whose interrupts to the OS are known to provide some amount of entropy.
- RANDOM_SWI: This software source refers to the entropy obtained during the execution of software interrupt handlers/threads in the system in response to Software Interrupts. Software interrupt handlers help to queue less critical processing outside of hardware interrupt handlers so that the hardware interrupt handlers perform minimal work.
- RANDOM_ATTACH: This software source is based on the time inside the OS at which a device-driver is attached to the associated devices in the system. This happens most during system boot-up.

7.1.3 SSH

63. 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***)

64. 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***)

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

65. 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. (***FTP_TRP.1/Admin, FCS_SSHS_EXT.1***)

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

RFC	Summary	TOE implementation of Security
RFC 4251	The Secure Shell (SSH) Protocol Architecture	**Host Keys:** The TOE uses an ECDSA Host Key for SSH v2, with a key size of 256 bits or greater, which is generated on initial setup of the TOE. It can be de-configured via the CLI and the key will be deleted and thus unavailable during connection establishment. This key is randomly generated to be unique to each TOE instance. The TOE presents the client with its public key and the client matches this key against its known_hosts list of keys. When a client connects to the TOE, the client will be able to determine if the same host key was used in previous connections, or if the key is different (per the SSHv2 protocol). Junos OS also supports RSA-based key establishment schemes with a key size of 2048 bits. **Policy Issues:** The TOE implements all mandatory algorithms and methods. The TOE can be configured to accept public-key based authentication and/or password-based authentication. The TOE does not require multiple authentication mechanisms for users. The TOE allows port forwarding and sessions to clients. The TOE has no X11 libraries or applications and X11 forwarding is prohibited. **Confidentiality:** The TOE does not accept the “none” cipher. supports AES-CBC-128, AES-CBC-256, AES-CTR-128, AES-CTR-256 encryption algorithms for protection of data over SSH and uses keys generated in accordance with “ecdsa-sha2-nistp256 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. Authentication succeeds if the correct private key is used. The TOE does not require multiple authentications (public key and password) for users. **Password Authentication Method:** The TOE supports password authentication. Expired passwords are not supported and cannot be used for authentication. **Host-Based Authentication:** The TOE does not support the configuration of host-based authentication methods.
RFC 4253	The Secure Shell (SSH) Transport Layer Protocol	**Encryption:** The TOE offers the following for encryption of SSH sessions: aes128-cbc and aes256-cbc, aes128-ctr, aes256-ctr. The TOE permits negotiation of encryption algorithms in each direction. The TOE does not allow the "none" algorithm for encryption. **Maximum Packet length:** Packets greater than 256Kbytes in an SSH transport connection are dropped and the connection is terminated by Junos OS. **Data Integrity:** The TOE permits negotiation of HMAC-SHA1 in each direction for SSH transport. **Key Exchange:** The TOE supports diffie-hellman-group14-sha1. **Key Re-Exchange:** The TOE performs a re-exchange when SSH_MSG_KEXINIT is received.

RFC	Summary	TOE implementation of Security
RFC 4254	Secure Shell (SSH) Connection Protocol	**Multiple channels:** The TOE assigns each channel a number (as detailed in RFC 4251, see above). **Data transfers:** The TOE supports a maximum window size of 256K bytes for data transfer. **Interactive sessions:** The TOE only supports interactive sessions that do NOT involve X11 forwarding. **Forwarded X11 connections:** This is not supported in the TOE. **Environment variable passing:** The TOE only sets variables once the server process has dropped privileges. **Starting shells/commands:** The TOE supports starting one of shell, application program or command (only one request per channel). These will be run in the context of a channel, and will not halt the execution of the protocol stack. **Window dimension change notices:** The TOE will accept notifications of changes to the terminal size (dimensions) from the client. **Port forwarding:** This is fully supported by the TOE.
RFC5656	SSH ECC Algorithm Integration	**ECDH Key Exchange**: The support key exchange methods specified in this RFC are ecdh-sha2-nistp256, ecdh-sha2-nistp384, or ecdh-sha2-nistp521. The client matches the key against its known_hosts list of keys. **Hashing**: Junos OS supports cryptographic hashing via the SHA-256, SHA-384 and SHA-512 algorithms, provided it has a message digest size of either 256, 384 or 512 bytes. **Required Curves:** All required curves are implemented: ecdh-sha2-nistp256, ecdh-sha2-nistp384, or ecdh-sha2-nistp521. None of the Recommended Curves are supported as they are not included in [NDcPP].
RFC 6668	sha2-Transport Layer Protocol	**Data Integrity Algorithms:** Both the recommended and optional algorithms hmac-sha2-256 and hmac-sha2-512 (respectively) are implemented for SSH transport.

Table 9 SSH RFC conformance

67. 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***).

7.2 Administrator Authentication

68. Junos OS enforces binding between human users and subjects. The Security Administrator[5] is responsible for provisioning user accounts, and only the Security Administrator can do so. (***FMT_SMR.2***)

69. 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.

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

- `login()`

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

- PAM Library module

71. 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.

72. 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).

73. 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.

74. 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.

75. 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). 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.

76. 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.

77. 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***)

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

- The password is hashed when stored, using hmac-sha1, sha256 or sha512.
- 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.

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

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

81. 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.

82. 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

83. 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.
- Authentication error – verifies that veriexec is enabled and operates as expected using /opt/sbin/kats/cannot-exec.real.
- Kernel, LibMD, OpenSSL,– verifies correct output from known answer tests for appropriate algorithms and verifies X509 certificate validity checks.

84. 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.

85. 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.

86. 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_TST_EXT.1***)

7.4 Trusted Update

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

88. The installable firmware package includes both the Junos OS VM and the WRL virtualization kernel. These cannot be updated separately in the evaluated configuration; they must be installed as a single package.

89. The installable firmware package has a digital signature that is checked when the Security Administrator attempts to install the package. The firmware is digitally signed, and provides a certificate chain which must terminate at one of the internal CA certificates. 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.

90. In the NDcPP deployment, "disable on-download-failure" is set to enforce revocation checks using a CRL in local trust store cache[6]. (An updated CRL is loaded during a firmware update, as it is embedded within the firmware binary.) If the certificate considered for validation is not present in the list of revoked certificates in the local cache, then the validation succeeds. If the CRL is not available in Junos OS cache, the certificate is considered to have failed validation. (***FIA_X509_EXT.2***)

91. 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 accompanying X.509 certificate (stored on the TOE filesystem in clear, protected by filesystem access rights). ECDSA (P-256) with SHA-256 is used for digit signature package verification.

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

93. A certificate may be loaded via command line, and is stored in SSD. Access to flash memory requires administrator credentials. Cryptographic keys are protected through the enforcement of kernel-level file access rights, limiting access to the contents of cryptographic key containers to processes with cryptographic rights. The TOE does not provide a CLI interface to permit the viewing of keys. (***FIA_X.509_EXT.1/Rev, FMT_MTD.1/CoreData***).

[6] The trust store, embedded in the running Junos, contains a number of CA certificates and their CRLs if applicable (a CRL is only present if that CA has ever revoked a certificate).

94. To validate X.509v3 certificates (as defined in RFC 5280) used to digitally sign the install packages, Junos OS extracts the subject, issuer, subjects public key, signature, basicConstraints and validity period fields. If any of those fields is not present, the validation fails.

95. Junos OS 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).

96. 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. 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.

97. Junos OS validates a certificate path by building a chain of 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. The certificate chains are complete - up to the rootCA - which is then verified against the internal trust store[7].

98. (***FCS_COP.1/SigGen, FPT_TUD_EXT.2, FIA_X509_EXT.1/Rev, FMT_MTD.1/CoreData, FIA_X509_EXT.2, FPT_TUD_EXT.2***)

7.5 Audit

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

- Start-up and shut-down of the audit functions
- Administrative login and logout
- Configuration is committed
- Configuration is changed (includes all management activities of TSF data)
- Generating/import of, changing, or deleting of cryptographic keys (see below for more detail)
- Resetting passwords
- Starting and stopping services
- All use of the identification and authentication mechanisms
- Unsuccessful login attempts limit is met or exceeded
- Any attempt to initiate a manual update
- Result of the update attempt (success or failure)
- The termination of a local/remote/interactive session by the session locking mechanism
- Initiation/termination/failure of the SSH trusted channel to syslog server
- Initiation/termination/failure of the SSH trusted path with Admin

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

- configure the access banner;

[7] The last certificate in the provided chain must match a certificate in the internal trust store, or be issued by one.

- 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.

101. 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.

102. 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).

103. 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):

- SSH session keys– key reference provided by process id
- 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

104. 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
```

105. 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).

106. All events recorded by syslog are timestamped. The clock function of Junos OS provides a source of date and time information for the appliance, used in audit timestamps. Wind River Linux kernel provides the current time when it bootstraps the Junos OS VM. Once the Junos OS VM is

started it maintains its own time using the hardware Time Stamp Counter as the clock source[8]. (***FAU_GEN.2, FPT_STM_EXT.1***)

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

108. 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.

109. 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

110. 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***)

111. 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***)

112. 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***)

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

  - 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***)
  - 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)
- Perform management functions (***FMT_SMF.1***):
  - Configure the access banner (***FTA_TAB.1***)
  - Configure the session inactivity time before session termination or locking, including termination of session when serial console cable is disconnected (***FTA_SSL_EXT.1, FTA_SSL.3***)
  - Import X.509v3 certificates to the TOE's trust store (via the Trusted update process)
  - 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***)

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

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	FPT_STM.1	FPT_STM_EXT.1 included (which is hierarchical to FPT_STM.1)
FAU_GEN.2	FAU_GEN.1 FIA_UID.1	FAU_GEN.1 Included Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator identification timing
FAU_STG_EXT.1	FAU_GEN.1 FTP_ITC.1	FAU_GEN.1 included FTP_ITC.1 included
FAU_STG.1	FAU_GEN.1	FAU_GEN.1 Included
FCS_CKM.1	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
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

Security Functional Requirement	Dependency	Rationale
FIA_AFL.1	FIA_UAU.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FIA_PMG_EXT.1	None	n/a
FIA_UIA_EXT.1	FTA_TAB.1	FTA_TAB.1 included
FIA_UAU_EXT.2	None	n/a
FIA_UAU.7	FIA_UAU.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FIA_X509_EXT.1/Rev	None	n/a
FIA_X509_EXT.2	None	n/a
FMT_MOF.1/ManualUpdate	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MOF.1/Services	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MOF.1/Functions	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MTD.1/CoreData	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_MTD.1/CryptoKeys	FMT_SMR.1 FMT_SMF.1	FMT_SMR.2 included FMT_SMF.1 included
FMT_SMF.1	None	n/a
FMT_SMR.2	FIA_UID.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FPT_SKP_EXT.1	None	n/a
FPT_APW_EXT.1	None	n/a
FPT_TST_EXT.1	None	n/a
FPT_TUD_EXT.1	FCS_COP.1/SigGen or FCS_COP.1/Hash	FCS_COP.1/SigGen
FPT_TUD_EXT.2	FPT_TUD_EXT.1	FPT_TUD_EXT.1 included
FPT_STM_EXT.1	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

Table 10 SFR Dependency Analysis

9 Glossary

AES	Advanced Encryption Standard
ANSI	American National Standards Institute
cPP	collaborative Protection Profile
CSP	Critical security parameter
DH	Diffie Hellman
EAL	Evaluation Assurance Level
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
EP	Extended Package, defined in [CC1]
FIPS	Federal Information Processing Standard
HMAC	Keyed-Hash Authentication Code
I&A	Identification and Authentication
ID	Identification
IP	Internet Protocol
ISO	International Organization for Standardization
IT	Information Technology
Junos	Juniper Operating System
KVM	Kernel-Based Virtual Machine
NDcPP	Network Device collaborative Protection Profile
NTP	Network Time Protocol
OSI	Open Systems Interconnect
OSP	Organizational Security Policy
PAM	Pluggable Authentication Module
PFE	Packet Forwarding Engine
PP	Protection Profile
RE	Routing Engine
RFC	Request for Comment
RNG	Random Number Generator
RSA	Rivest, Shamir, Adelman
SFP	Small Form-factor Pluggable
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SNMP	Simple Network Management Protocol
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF interfaces
WRL	Wind River Linux