IBM QRadar Security Intelligence Platform Version 7.5 Security Target

Version 0.8
June 26, 2023

Prepared for:

IBM Corporation

1701 North St., Bldg 256-1
Endicott, New York 13760

Prepared By:

Gossamer
Laboratories

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is IBM QRadar Security Intelligence Platform provided by IBM Corporation. The TOE is being evaluated as a network device.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
  - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a parenthetical number placed at the end of the component. For example FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement.
  - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
  - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
  - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – IBM QRadar Security Intelligence Platform Version 7.5 Security Target

ST Version – Version 0.8

ST Date – June 26, 2023

1.2 TOE Reference

TOE Identification – IBM Corporation IBM QRadar Security Intelligence Platform Version 7.5

TOE Developer – IBM Corporation

Evaluation Sponsor – IBM Corporation

1.3 TOE Overview

The Target of Evaluation (TOE) is IBM QRadar Security Intelligence Platform Version 7.5.

IBM QRadar Security Intelligence Platform is also known as the IBM QRadar Security Information and Event Management (SIEM). The QRadar SIEM is a network device intended to detect potential threats through the review of audit and event data collected from network sources. The TOE is administered either locally or remotely. The QRadar product consolidates log source event data from thousands of devices endpoints and applications distributed throughout a network.

1.4 TOE Description

IBM QRadar SIEM consolidates log source event data from device endpoints and applications that are distributed throughout a network. QRadar performs intermediate normalization and correlation activities on this raw data and can forward data to another network server when so configured. Communication with network peers for outbound log/event data is accomplished using TLS protected communication channels. QRadar is capable of providing an X.509v3 certificate to authenticate itself as part of an outbound TLS connection.

The IBM All-In-One: Dell 3128C, model utilizes an x86 64-bit CPU architecture, with 4 network interface cards, and varying amounts of memory. The TOE runs on the All-In-One device which uses an Intel® Xeon® Gold 5118 CPU without PAA.

The QRadar SIEM utilizes multiple cryptographic security kernel libraries internally, IBMFIPSJCE and OpenSSL. The related CAVP algorithm certs are shown in **Table 6-1** (OpenSSL) and **Table 6-2** (IBMFIPSJCE).

QRadar provides its cryptographic features through a Java implementation (IBMFIPSJCE) which is developed by IBM independently from QRadar. The OpenSSL library included in the TOE is OpenSSL 1.0.2k-fips as provided by RedHat Enterprise Linux. Thus, all cryptographic functions are provided by IBMFIPSJCE or OpenSSL.

1.4.1 TOE Architecture

The evaluated product is a single All-in-One device running QRadar SIEM with QFlow enabled. A QRadar QFlow collector collects network traffic passively through network taps and span ports. A QFlow collector can detect and collect information from networked applications. The All-in-One device is a self-contained appliance running the QRadar SIEM in a Red Hat RHEL 7.9 environment. The appliance makes only those interfaces offered by QRadar available.

The IBM All-In-One: Dell 3128C, model utilizes an x86 64-bit CPU architecture, with 4 network interface cards, and varying amounts of memory.

The All-In-One device can connect to an external audit server allowing QRadar to transmit audit and event data to an external server. All outbound audit data is transferred using TLS protected communication channels.

An IBM QRadar All-In-One device provides a trusted path to remote administrators using an HTTPS protected web GUI or SSH protected Command Line Interface (CLI). The QRadar system offers a CLI at the local console and remotely via SSH as an administrative interface. QRadar also offers a web interface for additional administrative functionality. A single device will have four (4) network connections which can be used either for remote management, receipt of event/syslog data, transmission of audit data, or other network support traffic (e.g.,DNS). A REST API interface is offered by QRadar and can be protected by HTTPS/TLS.

1.4.1.1 Physical Boundaries

The TOE is composed of one physical component that is accessed and managed by administrators from computers in the environment. The evaluated product is a single All-in-One device running QRadar SIEM with QFlow enabled.

The TOE can be configured to forward its audit records to an external syslog server in the environment. All audit records sent to the external syslog server are protected using a TLS channel.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by QRadar:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

1.4.1.2.1 Security audit

The TOE generates logs for a wide range of security relevant events. The TOE can be configured to store the logs locally so they can be accessed by an administrator and also to send the logs to a designated network peer using TLS to protect data while in transit. The TOE is also capable of acting as a log storage device and receiving TLS protected communication from network peers sending audit/event data.

1.4.1.2.2 Cryptographic support

The TOE utilizes NIST validated cryptographic algorithms to support key management, random bit generation, encryption/decryption, digital signature and secure hashing and key-hashing features in support of higher level cryptographic protocols including SSH and TLS.

1.4.1.2.3 Identification and authentication

The TOE requires users to be identified and authenticated before they can use functions mediated by the TOE, with the exception of reading the login banner. The TOE authenticates administrative users. In order for an administrative user to access the TOE, a user account including a user name and password must be created for the user.

1.4.1.2.4 Security management

The TOE provides an SSH protected Command Line Interface (CLI) commands and an HTTP over TLS (HTTPS) Graphical User Interface (GUI) to access the wide range of security management functions to manage its security policies. The TOE also offers HTTP over TLS protection for RESTAPI interfaces that can be used for administration. All administrative activity and functions including security management commands are limited to authorized users (i.e., administrators) only after they have provided acceptable user identification and authentication data to the TOE. The security management functions are controlled through the use of privileges associated with roles that can be assigned to TOE users.

1.4.1.2.5 Protection of the TSF

The TOE implements a number of features designed to protect itself to ensure the reliability and integrity of its security features.

It protects particularly sensitive data such as stored passwords and cryptographic keys so that they are not accessible even by an administrator. It also provides its own timing mechanism to ensure that reliable time information is available (e.g., for log accountability).

Note that the TOE is a single appliance, and as such, no intra-TOE communication is subject to any risks that may require special protection (e.g., cryptographic mechanisms).

The TOE includes functions to perform self-tests so that it might detect when it is failing. It also includes support to verify signatures on product updates. The signature of an update is verified against a known key for IBM which is installed in the TOE.

1.4.1.2.6 TOE access

The TOE can be configured to display a logon banner before a user session is established. The TOE also enforces inactivity timeouts for local and remote sessions.

1.4.1.2.7 Trusted path/channels

The TOE protects communication channels between itself and remote administrators using HTTPS/TLS and SSH. The SSH protocol is used to protect administrative connections utilizing the TOE's command line interface (CLI). Additionally, web-based GUI and RESTAPI interfaces are available for remote administration which are protected using HTTP over TLS (HTTPS/TLS).

The TOE also protects communication with network peers using TLS. Protected communication includes the TOE's outbound connection to an external audit server.

1.4.2 TOE Documentation

The TOE offers a series of documents that describe the installation process for the TOE as well as guidance for subsequent use and administration of the applicable security features. The following is a list of such documents.

- IBM QRadar Version 7.5 Common Criteria for NIAP Revision 1.0

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.
  - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017.
  - Part 3 Conformant
- Package Claims:
  - collaborative Protection Profile for Network Devices', Version 2.2e, 23 March 2020 (NDcPP22e)

Package	Technical Decision	Applied	Notes
CPP_ND_V2.2E	TD0670 - NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	Yes	
CPP_ND_V2.2E	TD0639 - NIT Technical Decision for Clarification for NTP MAC Keys	Yes	
CPP_ND_V2.2E	TD0638 - NIT Technical Decision for Key Pair Generation for Authentication	Yes	
CPP_ND_V2.2E	TD0636 - NIT Technical Decision for Clarification of Public Key User Authentication for SSH	No	Requirement not claimed
CPP_ND_V2.2E	TD0635 - NIT Technical Decision for TLS Server and Key Agreement Parameters	Yes	
CPP_ND_V2.2E	TD0634 - NIT Technical Decision for Clarification required for testing IPv6	Yes	
CPP_ND_V2.2E	TD0633 - NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	No	Requirement not claimed
CPP_ND_V2.2E	TD0632 - NIT Technical Decision for Consistency with Time Data for vNDs	Yes	
CPP_ND_V2.2E	TD0631 - NIT Technical Decision for Clarification of public key authentication for SSH Server	Yes	
CPP_ND_V2.2E	TD0592 - NIT Technical Decision for Local Storage of Audit Records	Yes	
CPP_ND_V2.2E	TD0591 - NIT Technical Decision for Virtual TOEs and hypervisors	Yes	
CPP_ND_V2.2E	TD0581 - NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	Yes	
CPP_ND_V2.2E	TD0580 - NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	Yes	
CPP_ND_V2.2E	TD0572 - NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes	
CPP_ND_V2.2E	TD0571 - NiT Technical Decision for Guidance on how to handle FIA_AFL.1	Yes	
CPP_ND_V2.2E	TD0570 - NiT Technical Decision for Clarification about FIA_AFL.1	Yes	
CPP_ND_V2.2E	TD0569 - NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	Yes	
CPP_ND_V2.2E	TD0564 - NiT Technical Decision for Vulnerability Analysis Search Criteria	Yes	
CPP_ND_V2.2E	TD0563 - NiT Technical Decision for Clarification of audit date information	Yes	
CPP_ND_V2.2E	TD0556 - NIT Technical Decision for RFC 5077 question	Yes	

Package	Technical Decision	Applied	Notes
CPP_ND_V2.2E	TD0555 - NIT Technical Decision for RFC Reference incorrect in TLSS Test	Yes	
CPP_ND_V2.2E	TD0547 - NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes	
CPP_ND_V2.2E	TD0546 - NIT Technical Decision for DTLS - clarification of Application Note 63	No	Requirement not claimed
CPP_ND_V2.2E	TD0538 - NIT Technical Decision for Outdated link to allowed-with list	Yes	
CPP_ND_V2.2E	TD0537 - NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	Yes	
CPP_ND_V2.2E	TD0536 - NIT Technical Decision for Update Verification Inconsistency	Yes	
CPP_ND_V2.2E	TD0528 - NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	No	Requirement not claimed
CPP_ND_V2.2E	TD0527 - Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	

2.1 Conformance Rationale

The ST conforms to the NDcPP22e. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the NDcPP22e and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The NDcPP22e offers additional information about the identified security objectives, but that has not been reproduced here and the NDcPP22e should be consulted if there is interest in that material.

In general, the NDcPP22e has defined Security Objectives appropriate for network devices and as such are applicable to the IBM QRadar Security Intelligence Platform TOE.

3.1 Security Objectives for the Operational Environment

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

OE.COMPONENTS_RUNNING (applies to distributed TOEs only)

For distributed TOEs, the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.

OE.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.

OE.NO_THRU_TRAFFIC_PROTECTION The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

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

OE.RESIDUAL_INFORMATION The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.

OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality.
For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.

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

OE.VM_CONFIGURATION (applies to vNDs only)
For vNDs, the Security Administrator ensures that the VS and VMs are configured to
- reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and
- correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting).

The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by virtualisation features such as cloning, save/restore, suspend/resume, and live migration.

If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDcPP22e. The NDcPP22e defines the following extended requirements and since they are not redefined in this ST the NDcPP22e should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- NDcPP22e:FAU_STG_EXT.1: Protected Audit Event Storage
- NDcPP22e:FCS_HTTPS_EXT.1: HTTPS Protocol
- NDcPP22e:FCS_RBG_EXT.1: Random Bit Generation
- NDcPP22e:FCS_SSHS_EXT.1: SSH Server Protocol - per TD0631
- NDcPP22e:FCS_TLSC_EXT.1: TLS Client Protocol Without Mutual Authentication - per TD0634
- NDcPP22e:FCS_TLSC_EXT.2: TLS Client Support for Mutual Authentication
- NDcPP22e:FCS_TLSS_EXT.1: TLS Server Protocol Without Mutual Authentication - per TD0635
- NDcPP22e:FIA_PMG_EXT.1: Password Management
- NDcPP22e:FIA_UAU_EXT.2: Password-based Authentication Mechanism
- NDcPP22e:FIA_UIA_EXT.1: User Identification and Authentication
- NDcPP22e:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
- NDcPP22e:FIA_X509_EXT.2: X.509 Certificate Authentication
- NDcPP22e:FIA_X509_EXT.3: X.509 Certificate Requests
- NDcPP22e:FPT_APW_EXT.1: Protection of Administrator Passwords
- NDcPP22e:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
- NDcPP22e:FPT_STM_EXT.1: Reliable Time Stamps - per TD0632
- NDcPP22e:FPT_TST_EXT.1: TSF testing
- NDcPP22e:FPT_TUD_EXT.1: Trusted update
- NDcPP22e:FTA_SSL_EXT.1: TSF-initiated Session Locking

5. Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs have all been drawn from the NDcPP22e. The refinements and operations already performed in the NDcPP22e are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP22e and any residual operations have been completed herein. Of particular note, the NDcPP22e made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the NDcPP22e. The NDcPP22e should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by IBM QRadar Security Intelligence Platform TOE.

Requirement Class	Requirement Component
FAU: Security audit	NDcPP22e:FAU_GEN.1: Audit Data Generation
	NDcPP22e:FAU_GEN.2: User identity association
	NDcPP22e:FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	NDcPP22e:FCS_CKM.1: Cryptographic Key Generation
	NDcPP22e:FCS_CKM.2: Cryptographic Key Establishment
	NDcPP22e:FCS_CKM.4: Cryptographic Key Destruction
	NDcPP22e:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	NDcPP22e:FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	NDcPP22e:FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	NDcPP22e:FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	NDcPP22e:FCS_HTTPS_EXT.1: HTTPS Protocol
	NDcPP22e:FCS_RBG_EXT.1: Random Bit Generation
	NDcPP22e:FCS_SSHS_EXT.1: SSH Server Protocol - per TD0631
	NDcPP22e:FCS_TLSC_EXT.1: TLS Client Protocol Without Mutual Authentication - per TD0634
	NDcPP22e:FCS_TLSC_EXT.2: TLS Client Support for Mutual Authentication
	NDcPP22e:FCS_TLSS_EXT.1: TLS Server Protocol Without Mutual Authentication - per TD0635
FIA: Identification and authentication	NDcPP22e:FIA_AFL.1: Authentication Failure Management
	NDcPP22e:FIA_PMG_EXT.1: Password Management
	NDcPP22e:FIA_UAU.7: Protected Authentication Feedback
	NDcPP22e:FIA_UAU_EXT.2: Password-based Authentication Mechanism
	NDcPP22e:FIA_UIA_EXT.1: User Identification and Authentication
	NDcPP22e:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	NDcPP22e:FIA_X509_EXT.2: X.509 Certificate Authentication
	NDcPP22e:FIA_X509_EXT.3: X.509 Certificate Requests
FMT: Security management	NDcPP22e:FMT_MOF.1/ManualUpdate: Management of security functions behaviour
	NDcPP22e:FMT_MTD.1/CoreData: Management of TSF Data
	NDcPP22e:FMT_MTD.1/CryptoKeys: Management of TSF Data
	NDcPP22e:FMT_SMF.1: Specification of Management Functions - per TD0631
	NDcPP22e:FMT_SMR.2: Restrictions on Security Roles

Requirement Class	Requirement Component
FPT: Protection of the TSF	NDcPP22e:FPT_APW_EXT.1: Protection of Administrator Passwords
	NDcPP22e:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	NDcPP22e:FPT_STM_EXT.1: Reliable Time Stamps - per TD0632
	NDcPP22e:FPT_TST_EXT.1: TSF testing
	NDcPP22e:FPT_TUD_EXT.1: Trusted update
FTA: TOE access	NDcPP22e:FTA_SSL.3: TSF-initiated Termination
	NDcPP22e:FTA_SSL.4: User-initiated Termination
	NDcPP22e:FTA_SSL_EXT.1: TSF-initiated Session Locking
	NDcPP22e:FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	NDcPP22e:FTP_ITC.1: Inter-TSF trusted channel
	NDcPP22e:FTP_TRP.1/Admin: Trusted Path

Table 5-1 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (NDcPP22e:FAU_GEN.1)

NDcPP22e: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).
- [***no other actions***];

d) Specifically defined auditable events listed in **Table 5-2**.

Requirement	Auditable Events	Additional Content
NDcPP22e:FAU_GEN.1	None	None
NDcPP22e:FAU_GEN.2	None	None
NDcPP22e:FAU_STG_EXT.1	None	None
NDcPP22e:FCS_CKM.1	None	None
NDcPP22e:FCS_CKM.2	None	None
NDcPP22e:FCS_CKM.4	None	None
NDcPP22e:FCS_COP.1/DataEncryption	None	None
NDcPP22e:FCS_COP.1/Hash	None	None
NDcPP22e:FCS_COP.1/KeyedHash	None	None
NDcPP22e:FCS_COP.1/SigGen	None	None
NDcPP22e:FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
NDcPP22e:FCS_RBG_EXT.1	None	None
NDcPP22e:FCS_SSHS_EXT.1	Failure to establish an SSH session.	Reason for failure.
NDcPP22e:FCS_TLSC_EXT.1	Failure to establish a TLS Session.	Reason for failure.

Requirement	Auditable Events	Additional Content
NDcPP22e:FCS_TLSC_EXT.2	None	None
NDcPP22e:FCS_TLSS_EXT.1	None	None
NDcPP22e:FIA_AFL.1	Unsuccessful login attempt limit is met or exceeded.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_PMG_EXT.1	None	None
NDcPP22e:FIA_UAU.7	None	None
NDcPP22e:FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate. Any addition, replacement or removal of trust anchors in the TOE's trust store	Reason for failure of certificate validation Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
NDcPP22e:FIA_X509_EXT.2	None	None
NDcPP22e:FIA_X509_EXT.3	None	None
NDcPP22e:FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	None
NDcPP22e:FMT_MTD.1/CoreData	None	None
NDcPP22e:FMT_MTD.1/CryptoKeys	None	None
NDcPP22e:FMT_SMF.1	All management activities of TSF data.	None
NDcPP22e:FMT_SMR.2	None	None
NDcPP22e:FPT_APW_EXT.1	None	None
NDcPP22e:FPT_SKP_EXT.1	None	None
NDcPP22e: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).
NDcPP22e:FPT_TST_EXT.1	None	None
NDcPP22e:FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None
NDcPP22e:FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None
NDcPP22e:FTA_SSL.4	The termination of an interactive session.	None
NDcPP22e:FTA_SSL_EXT.1	(if 'lock the session' is selected) Any attempts at unlocking of an interactive session. (if 'terminate the session' is selected) The termination of a local session by the session locking mechanism.	None
NDcPP22e:FTA_TAB.1	None	None
NDcPP22e:FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted	Identification of the initiator and target of failed trusted

Requirement	Auditable Events	Additional Content
	channel. Failure of the trusted channel functions.	channels establishment attempt.
NDcPP22e:FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	None

Table 5-2 Auditable Events

NDcPP22e: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 cPP/ST, information specified in column three of **Table 5-2**.

5.1.1.2 User identity association (NDcPP22e:FAU_GEN.2)

NDcPP22e: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.

5.1.1.3 Protected Audit Event Storage (NDcPP22e:FAU_STG_EXT.1)

NDcPP22e: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.

NDcPP22e:FAU_STG_EXT.1.2

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

NDcPP22e:FAU_STG_EXT.1.3

The TSF shall [***overwrite previous audit records according to the following rule: [overwrite audit data using a log file rotation which deletes the oldest archived log file]***] when the local storage space for audit data is full.

5.1.2 Cryptographic support (FCS)

5.1.2.1 Cryptographic Key Generation (NDcPP22e:FCS_CKM.1)

NDcPP22e: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' [selection: P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4,***
- ***FFC Schemes using 'safe-prime' groups that meet the following: 'NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' and [RFC 3526]***].

5.1.2.2 Cryptographic Key Establishment (NDcPP22e:FCS_CKM.2)

NDcPP22e:FCS_CKM.2.1

The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- ***RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1 ,***
- ***Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' (TD0581 applied),***
- ***FFC Schemes using 'safe-prime' groups that meet the following: 'NIST Special Publication 800-56A Revision 3, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' and [groups listed in RFC 3526] (TD0580 applied)***].

5.1.2.3 Cryptographic Key Destruction (NDcPP22e:FCS_CKM.4)

NDcPP22e: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 [a new value of the key]***]

that meets the following: No Standard.

5.1.2.4 Cryptographic Operation (AES Data Encryption/Decryption) (NDcPP22e:FCS_COP.1/DataEncryption)

NDcPP22e:FCS_COP.1.1/DataEncryption

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

5.1.2.5 Cryptographic Operation (Hash Algorithm) (NDcPP22e:FCS_COP.1/Hash)

NDcPP22e:FCS_COP.1.1/Hash

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

5.1.2.6 Cryptographic Operation (Keyed Hash Algorithm) (NDcPP22e:FCS_COP.1/KeyedHash)

NDcPP22e:FCS_COP.1.1/KeyedHash

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

5.1.2.7 Cryptographic Operation (Signature Generation and Verification) (NDcPP22e:FCS_COP.1/SigGen)

NDcPP22e: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]]

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].

5.1.2.8 HTTPS Protocol (NDcPP22e:FCS_HTTPS_EXT.1)

NDcPP22e:FCS_HTTPS_EXT.1.1
The TSF shall implement the HTTPS protocol that complies with RFC 2818.

NDcPP22e:FCS_HTTPS_EXT.1.2
The TSF shall implement HTTPS using TLS.

NDcPP22e:FCS_HTTPS_EXT.1.3
If a peer certificate is presented, the TSF shall [***not establish the connection***] if the peer certificate is deemed invalid.

5.1.2.9 Random Bit Generation (NDcPP22e:FCS_RBG_EXT.1)

NDcPP22e:FCS_RBG_EXT.1.1
The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [***CTR_DRBG (AES), HASH_DRBG(AES)***].

NDcPP22e:FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[one] software-based noise source***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.2.10 SSH Server Protocol - per TD0631 (NDcPP22e:FCS_SSHS_EXT.1)

NDcPP22e:FCS_SSHS_EXT.1.1
The TSF shall implement the SSH protocol that complies with: RFC(s) 4251, 4252, 4253, 4254, [***4256, 4344, 5647, 5656, 6187, 6668, 8268, 8308 section 3.1, 8332***].

NDcPP22e:FCS_SSHS_EXT.1.2
The TSF shall ensure that the SSH protocol implementation supports the following user authentication methods as described in RFC 4252: public key-based, [***password-based***].

NDcPP22e:FCS_SSHS_EXT.1.3
The TSF shall ensure that, as described in RFC 4253, packets greater than [***262127***] bytes in an SSH transport connection are dropped.

NDcPP22e: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***].

NDcPP22e:FCS_SSHS_EXT.1.5
The TSF shall ensure that the SSH public-key based authentication implementation uses [***ssh-rsa, rsa-sha2-256, rsa-sha2-512***] as its public key algorithm(s) and rejects all other public key algorithms.

NDcPP22e:FCS_SSHS_EXT.1.6
The TSF shall ensure that the SSH transport implementation uses [***hmac-sha2-256, hmac-sha2-512***] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

NDcPP22e:FCS_SSHS_EXT.1.7
The TSF shall ensure that [***diffie-hellman-group14-sha1***] and [***diffie-hellman-group14-sha256***] are the only allowed key exchange methods used for the SSH protocol.

NDcPP22e:FCS_SSHS_EXT.1.8
The TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

5.1.2.11 TLS Client Protocol Without Mutual Authentication - per TD0634 (NDcPP22e:FCS_TLSC_EXT.1)

NDcPP22e:FCS_TLSC_EXT.1.1

The TSF shall implement [***TLS 1.2 (RFC 5246)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [

- ***TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,***
- ***TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,***

]

and no other ciphersuites.

NDcPP22e:FCS_TLSC_EXT.1.2

The TSF shall verify that the presented identifier matches [***the reference identifier per RFC 6125 section 6***].

NDcPP22e:FCS_TLSC_EXT.1.3

When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [***Not implement any administrator override mechanism***].

NDcPP22e:FCS_TLSC_EXT.1.4

The TSF shall [***present the Supported Elliptic Curves/Supported Groups Extension with the following curves/groups: [secp256r1, secp384r1, secp521r1]***] in the Client Hello.

5.1.2.12 TLS Client Support for Mutual Authentication (NDcPP22e:FCS_TLSC_EXT.2)

NDcPP22e:FCS_TLSC_EXT.2.1

The TSF shall support TLS communication with mutual authentication using X.509v3 certificates.

5.1.2.13 TLS Server Protocol Without Mutual Authentication - per TD0635 (NDcPP22e:FCS_TLSS_EXT.1)

NDcPP22e:FCS_TLSS_EXT.1.1

The TSF shall implement [***TLS 1.2(RFC 5246)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [

- ***TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,***
- ***TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,***

]

and no other ciphersuites.

NDcPP22e:FCS_TLSS_EXT.1.2

The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [***TLS 1.1***].

NDcPP22e:FCS_TLSS_EXT.1.3

The TSF shall perform key establishment for TLS using [

- ***Diffie-Hellman parameters with size [2048 bits],***
- ***ECDHE curves [secp256r1, secp384r1, secp521r1] and no other curves***].

NDcPP22e:FCS_TLSS_EXT.1.4

The TSF shall support [***session resumption based on session tickets according to RFC 5077***].

5.1.3 Identification and authentication (FIA)

5.1.3.1 Authentication Failure Management (NDcPP22e:FIA_AFL.1)

NDcPP22e:FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within [**the range of 1 to 2,147,483,647**] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password.

NDcPP22e:FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met, the TSF shall [***prevent the offending Administrator from successfully establishing a remote session using any***

authentication method that involves a password until an Administrator defined time period has elapsed].

5.1.3.2 Password Management (NDcPP22e:FIA_PMG_EXT.1)

NDcPP22e: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: [***'!', '@', '#', '$', '%', '^', '&', '*', '(', ')'***];
b) Minimum password length shall be configurable to between [**15**] and [**255**] characters.

5.1.3.3 Protected Authentication Feedback (NDcPP22e:FIA_UAU.7)

NDcPP22e: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.

5.1.3.4 Password-based Authentication Mechanism (NDcPP22e:FIA_UAU_EXT.2)

NDcPP22e:FIA_UAU_EXT.2.1
The TSF shall provide a local [***password-based, SSH public key-based***] authentication mechanism to perform local administrative user authentication.

5.1.3.5 User Identification and Authentication (NDcPP22e:FIA_UIA_EXT.1)

NDcPP22e:FIA_UIA_EXT.1.1
The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

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

NDcPP22e: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.

5.1.3.6 X.509 Certificate Validation (NDcPP22e:FIA_X509_EXT.1/Rev)

NDcPP22e:FIA_X509_EXT.1.1/Rev
The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [***a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3***]
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
    - 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.

NDcPP22e: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.

5.1.3.7 X.509 Certificate Authentication (NDcPP22e:FIA_X509_EXT.2)

NDcPP22e:FIA_X509_EXT.2.1

The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [***TLS***], and [***no additional uses***].

NDcPP22e:FIA_X509_EXT.2.2

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

5.1.3.8 X.509 Certificate Requests (NDcPP22e:FIA_X509_EXT.3)

NDcPP22e:FIA_X509_EXT.3.1

The TSF shall generate a Certification Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [***Common Name, Organization, Organizational Unit, Country***].

NDcPP22e:FIA_X509_EXT.3.2

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

5.1.4 Security management (FMT)

5.1.4.1 Management of security functions behaviour (NDcPP22e:FMT_MOF.1/ManualUpdate)

NDcPP22e:FMT_MOF.1.1/ManualUpdate

The TSF shall restrict the ability to enable the functions to perform manual updates to Security Administrators.

5.1.4.2 Management of TSF Data (NDcPP22e:FMT_MTD.1/CoreData)

NDcPP22e:FMT_MTD.1.1/CoreData

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

5.1.4.3 Management of TSF Data (NDcPP22e:FMT_MTD.1/CryptoKeys)

NDcPP22e:FMT_MTD.1.1/CryptoKeys

The TSF shall restrict the ability to manage the cryptographic keys to Security Administrators.

5.1.4.4 Specification of Management Functions - per TD0631 (NDcPP22e:FMT_SMF.1)

NDcPP22e: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 behavior (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full),***
- ***Ability to manage the cryptographic keys,***
- ***Ability to configure the cryptographic functionality,***
- ***Ability to configure thresholds for SSH rekeying,***

- ***Ability to set the time which is used for time-stamps;***
- ***Ability to configure the reference identifier for the peer;***
- ***Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors,***
- ***Ability to import X509v3 certificates to the TOE's trust store***].

5.1.4.5 Restrictions on Security Roles (NDcPP22e:FMT_SMR.2)

NDcPP22e:FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

NDcPP22e:FMT_SMR.2.2

The TSF shall be able to associate users with roles.

NDcPP22e: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.

5.1.5 Protection of the TSF (FPT)

5.1.5.1 Protection of Administrator Passwords (NDcPP22e:FPT_APW_EXT.1)

NDcPP22e:FPT_APW_EXT.1.1

The TSF shall store administrative passwords in non-plaintext form.

NDcPP22e:FPT_APW_EXT.1.2

The TSF shall prevent the reading of plaintext administrative passwords.

5.1.5.2 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) (NDcPP22e:FPT_SKP_EXT.1)

NDcPP22e:FPT_SKP_EXT.1.1

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

5.1.5.3 Reliable Time Stamps - per TD0632 (NDcPP22e:FPT_STM_EXT.1)

NDcPP22e:FPT_STM_EXT.1.1

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

NDcPP22e:FPT_STM_EXT.1.2

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

5.1.5.4 TSF testing (NDcPP22e:FPT_TST_EXT.1)

NDcPP22e:FPT_TST_EXT.1.1

The TSF shall run a suite of the following self-tests [***during initial start-up (on power on), periodically during normal operation***] to demonstrate the correct operation of the TSF: [**cryptographic known answer self-tests run at startup and TOE software integrity checks run periodically**].

5.1.5.5 Trusted update (NDcPP22e:FPT_TUD_EXT.1)

NDcPP22e: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***].

NDcPP22e: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***].

NDcPP22e:FPT_TUD_EXT.1.3

The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature***] prior to installing those updates.

5.1.6 TOE access (FTA)

5.1.6.1 TSF-initiated Termination (NDcPP22e:FTA_SSL.3)

NDcPP22e:FTA_SSL.3.1

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

5.1.6.2 User-initiated Termination (NDcPP22e:FTA_SSL.4)

NDcPP22e:FTA_SSL.4.1

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

5.1.6.3 TSF-initiated Session Locking (NDcPP22e:FTA_SSL_EXT.1)

NDcPP22e:FTA_SSL_EXT.1.1

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

5.1.6.4 Default TOE Access Banners (NDcPP22e:FTA_TAB.1)

NDcPP22e: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.

5.1.7 Trusted path/channels (FTP)

5.1.7.1 Inter-TSF trusted channel (NDcPP22e:FTP_ITC.1)

NDcPP22e:FTP_ITC.1.1

The TSF shall be capable of using [***TLS***] 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.

NDcPP22e:FTP_ITC.1.2

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

NDcPP22e:FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [**export to an audit server**].

5.1.7.2 Trusted Path (NDcPP22e:FTP_TRP.1/Admin)

NDcPP22e:FTP_TRP.1.1/Admin

The TSF shall be capable of using [***SSH, TLS, HTTPS***] 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.

NDcPP22e:FTP_TRP.1.2/Admin

The TSF shall permit remote Administrators to initiate communication via the trusted path.

NDcPP22e:FTP_TRP.1.3/Admin

The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administration actions.

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic Functional Specification
AGD: Guidance documents	AGD_OPE.1: Operational User Guidance
	AGD_PRE.1: Preparative Procedures
ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM Coverage
ATE: Tests	ATE_IND.1: Independent Testing - Conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability Survey

Table 5-3 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic Functional Specification (ADV_FSP.1)

ADV_FSP.1.1d

The developer shall provide a functional specification.

ADV_FSP.1.2d

The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.1.1c

The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2c

The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3c

The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4c

The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e

The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1d

The developer shall provide operational user guidance.

AGD_OPE.1.1c

The operational user guidance shall describe, for each user role, the user accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c

The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c

The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c

The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c

The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.

AGD_OPE.1.6c

The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c

The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d

The developer shall provide the TOE, including its preparative procedures.

AGD_PRE.1.1c

The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c

The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e

The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d

The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c

The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3.2 TOE CM Coverage (ALC_CMS.1)

ALC_CMS.1.1d

The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c

The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2c

The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.4 Tests (ATE)

5.2.4.1 Independent Testing - Conformance (ATE_IND.1)

ATE_IND.1.1d

The developer shall provide the TOE for testing.

ATE_IND.1.1c

The TOE shall be suitable for testing.

ATE_IND.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2e

The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability Survey (AVA_VAN.1)

AVA_VAN.1.1d

The developer shall provide the TOE for testing.

AVA_VAN.1.1c

The TOE shall be suitable for testing.

AVA_VAN.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2e

The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3e

The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6. TOE Summary Specification

This chapter describes the security functions:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

6.1 Security audit

The TOE acts as a standalone network device generating audit data. The TOE can be configured to forward real-time audit data to a remote network device. This forwarding can include its own audit data. Thus, the TOE can be configured to send its own audit data to an external network peer using a trusted channel protected by TLS.

All audit logs are stored in plain text and are archived and compressed when the audit log file reaches 200 MB. The current log file is named audit.log. When the file reaches 200 MB, the file is compressed and renamed to audit.log.1.gz. The file number increments each time that a log file is archived. QRadar stores up to 50 archived log files. These audit files are accessible only to authenticated administrators.

The Security audit function satisfies the following security functional requirements:

- NDcPP22e:FAU_GEN.1: Auditing is always being generated when the TOE is running. Thus, the start and stop of auditing corresponds to boot and shutdown of the system. The boot and shutdown of the system are audited. The TOE audits the events identified in **Table 5-2 Auditable Events**.

  Audit records generated by the TOE include a date and time, event type, success or failure indication. For actions that can be assigned to a specific user or network entity, a subject identifier (e.g., username or network address) in included within the audit record. Audit records identified by **Table 5-2 Auditable Events** contain the additional information described by column 3 of the same table. Audit records contain a certificate's entire domain name to identify the certificate.

- NDcPP22e:FAU_GEN.2: For actions that can be assigned to a specific user or network entity, a subject identifier (e.g., username or network address) in included within the audit record.

- NDcPP22e:FAU_STG_EXT.1: The TOE can store audit data as well as transmit audit data in real time to an external audit server through a trusted channel protected by TLS. The external audit server can be any server supporting the transmission of audit data tunneled within TLS. All audit logs when stored internally are stored in plain text. The authorized administrator can only view audit records stored locally. The TOE provides the administrator this ability to view audit records only through the WebUI. These logs are archived and compressed when the audit log file size reaches 200 MB. The current log file is named audit.log. When the file reaches 200 MB, the file is compressed and renamed to audit.log.1.gz. The file number increments each time that a log file is archived. QRadar stores up to 50 archived log files. The TOE deletes the oldest archived log file, renumbering existing files when a new file is to be archived.

6.2 Cryptographic support

The TOE utilizes cryptographic support from RedHat's OpenSSL (1.0.2k-FIPS) and IBM FIPS JCE (version 1.8) library. Both are multi-algorithm libraries providing general-purpose cryptographic services. The purpose of the OpenSSL library is to provide an API for cryptographic functionality for system services. The OpenSSL library also provides centralized control over FIPS-Approved mode status, provides availability of only CAVP/FIPS-Approved algorithms or vendor-affirmed implementations of non FIPS-Approved algorithms, and provides for centralized logging and reporting of the cryptographic engine.

The purpose of the IBM FIPS JCE library is to provide only CAVP verified algorithms for use in Java based processes within the TOE. The TOE utilizes these libraries for cryptographic operations it performs (including those associated with TLS, certificate operations, system integrity, and update verification). The libraries used by the TOE have obtained CAVP certificates as shown in the following tables.

The TOE utilizes the IBM JCE library for cryptographic operations surrounding the trusted channel for TLS communication to a syslog server. This includes cryptographic operations for signature generation, signature verification, key establishment, random number generation, encryption, decryption, and hashing.

The TOE also uses the OpenSSL library to provide the same cryptographic operations in support of the TOE offered trusted paths via the HTTPS/TLS protected Web UI and CLI protected SSH. Additionally, the TOE also uses the OpenSSL library for the generation of keys during creation of a Certificate Signing Request (CSR). The TOE uses the OpenSSL library to perform verification of RSA signatures on product updates in support of FPT_TUD_EXT.1.

The TOE runs on the QRadar 3128C Appliance, which includes an Intel® Xeon® Gold 5118 CPU without PAA and operating system support from Red Hat Enterprise Linux server release 7.9.

Functions	Standards	SFR	CAVP Certificate
Encryption/Decryption			
AES CBC (128 and 256 bits) AES GCM (128 bits)	ISO 18033-3 ISO 10116 ISO 19772 FIPS Pub 197 NIST SP 800-38A	FCS_COP.1/DataEncryption	A2426
Cryptographic hashing			
SHA-256/512	FIPS Pub 180-4 ISO/IEC 10118-3:2004	FCS_COP.1/Hash	A2426
Keyed-hash message authentication			
HMAC_SHA-256, HMAC-SHA-512 (digest sizes and block sizes of 256 and 512 bits)	FIPS Pub 198-1 FIPS Pub 180-4 ISO/IEC 9797-2:2011	FCS_COP.1/KeyedHash	A2426
Cryptographic Signature Services			
RSA 2048-bit Signature Gen & Verify w/ SHA-256 and SHA-512	FIPS Pub 186-4 ISO/IEC 9796-2	FCS_COP.1/SigGen	A2426
Random bit generation			
CTR_DRBG (AES)	FIPS SP 800-90B ISO/IEC 18031:2011	FCS_RBG_EXT.1	A2426
Cryptographic Key Generation			
RSA 2048-bit Key Gen	FIPS Pub 186-4 ISO/IEC 9796-2	FCS_CKM.1	A2426
ECDSA P-256, P-384, P-521 Key Gen	FIPS PUB 186-4	FCS_CKM.1	A2426
Key Establishment			
RSA	RSAES-PKCS1-v1_5	FCS_CKM.2	Vendor Affirmed
KAS ECC	FIPS Pub 800-56A Rev 3	FCS_CKM.2	A2426
FFC Schemes using safe-prime groups Diffie-Hellman Group 14	NIST SP 800-56A Rev 3	FCS_CKM.2	Vendor Affirmed

Table 6-1 CAVP Certificates for OpenSSL library

Functions	Standards	SFR	CAVP Certificate
Encryption/Decryption			
AES CBC (128 and 256 bits) AES GCM (128 bits)	ISO 18033-3 ISO 10116 ISO 19772 FIPS Pub 197 NIST SP 800-38A	FCS_COP.1/DataEncryption	A2425
Cryptographic hashing			
SHA-256/512	FIPS Pub 180-4 ISO/IEC 10118-3:2004	FCS_COP.1/Hash	A2425
Keyed-hash message authentication			
HMAC_SHA-256, HMAC-SHA-512 (digest sizes and block sizes of 256 and 512 bits)	FIPS Pub 198-1 FIPS Pub 180-4 ISO/IEC 9797-2:2011	FCS_COP.1/KeyedHash	A2425
Cryptographic Signature Services			
RSA 2048-bit Signature Gen & Verify w/ SHA-256 and SHA-512	FIPS Pub 186-4 ISO/IEC 9796-2	FCS_COP.1/SigGen	A2425
Random bit generation			
HASH_DRBG	FIPS SP 800-90B ISO/IEC 18031:2011	FCS_RBG_EXT.1	A2425
Cryptographic Key Generation			
RSA 2048-bit Key Gen	FIPS Pub 186-4 ISO/IEC 9796-2	FCS_CKM.1	A2425
ECDSA P-256, P-384, P-521 Key Gen	FIPS Pub 186-4	FCS_CKM.1	A2425
Key Establishment			
RSA	RSAES-PKCS1-v1_5	FCS_CKM.2	Vendor Affirmed
KAS ECC	FIPS Pub 800-56A Rev 3	FCS_CKM.2	A2425
FFC Schemes using safe-prime groups Diffie-Hellman Group 14	NIST SP 800-56A Rev 3	FCS_CKM.2	Vendor Affirmed

Table 6-2 CAVP Certificates for IBMFIPSJCE library

These supporting cryptographic functions are provided by the TOE to support the SSHv2 (compliant with RFCs 4251, 4252, 4253, and 4254), and TLSv1.2 (compliant with RFC 5289 to communicate to an audit server and remote administrator) secure communication protocols. No older version of TLS or of SSL are supported. The TOE supports the TLS ciphersuites identified below. The TOE also supports Diffie-Hellman key group 14 (2048 bit) key generation for SSH key establishment.

The TOE supports the following TLS ciphersuites to connect with external syslog audit servers (provided by the IBMFIPSJCE library).

- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

The TOE allows inbound TLS connections using only the following ciphersuites (provided by OpenSSL library).

- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

For both TLS inbound and outbound connections the TOE utilizes RSA w/ key sizes of 2048-bits for authentication and can support ECDHE key exchanges. It also indicates that by default the TOE supports Elliptic Curves with secp256r1, secp384r1, secp521r1 in TLS ECDHE key exchanges.

The TOE supports the secret keys, private keys and CSPs as shown in **Table 6-3 Cryptographic Keys**.

Key or CSP:	Zeroized upon:	Stored in:	Zeroized by:
SSH host RSA private key	Administrative action	/root/.ssh/id_rsa	Generating new key, or uninstalling the module
SSH host RSA public key	Administrative action	/root/.ssh/id_rsa.pub	Generating new key, or uninstalling the module
SSH session key	Connection Termination	RAM	API call, power cycle, or host reboot
TLS host RSA private key	Administrative action	/etc/httpd/conf/certs/cert.key	Generating new private key, or uninstalling the module
TLS host RSA digital certificate	Administrative action	/etc/httpd/conf/certs/cert.cert	Generating new certificate, or uninstalling the module
TLS pre-master secret	Process Restart (Manual)	RAM	API call, power cycle, or host reboot
TLS session key	Process Restart	RAM	API call, power cycle, or host reboot
Administrator/User Passwords	User-driven account management	On disk, in hashed form	Setting new password, or uninstalling the module
GPG Update key	Install/Setup and Updates	Hard coded, internal to shared library	Uninstalling the module
DRBG Seed	System Reboot	RAM	API call, power cycle, or host reboot
DRBG Value V	System Reboot	RAM	API call, power cycle, or host reboot
DRBG Constant C	System Reboot	RAM	API call, power cycle, or host reboot

Table 6-3 Cryptographic Keys

The TOE stores all persistent secret and private keys on disk and stores all ephemeral keys in RAM (as indicated in the above table). Additionally, the TOE is designed to zeroize secret and private keys when they are no longer required by the TOE as detailed below. The TOE's zeroization has been subjected to FIPS 140 validation. Note that zeroization occurs as shown in column 4 of **Table 6-3**.

The Cryptographic support function satisfies the following security functional requirements:

- NDcPP22e:FCS_CKM.1: The TOE generates RSA keys of size 2048-bits using the Openssl library outlined in **Table 6-1 CAVP Certificates for OpenSSL library** and **Table 6-2 CAVP Certificates for IBMFIPSJCE library.** The TOE also supports FFC w/ safe-primes using Diffie-Hellman key group 14 (2048 bit) key generation for SSH key establishment. The TOE supports ECC key generation as part of TLS using NIST curves P-256, P-384, P-521 which are not configurable by administrators.

- NDcPP22e:FCS_CKM.2: The TOE generates RSA keys for use with both inbound and outbound TLS using key establishment schemes identified by **Table 6-1 CAVP Certificates for OpenSSL library** and **Table 6-2 CAVP Certificates for IBMFIPSJCE library**. The TOE also supports a Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526, Section 3. The TOE implements Diffie-Hellman group 14, using the modulus and generator specified by Section 3 of RFC3526.

Scheme	SFR	Service
FFC Schemes using 'safe-prime'	FCS_SSHS_EXT.1	Remote Administration
ECDHE	FCS_TLSC_EXT.1	Audit Server communication
ECDHE	FCS_TLSS_EXT.1	Remote Administration
RSA	FCS_SSHS_EXT.1	Remote Administration

Table 6-4 Key Establishment Methods

- NDcPP22e:FCS_CKM.4: The TOE stores all persistent secret and private keys on disk and stores all ephemeral keys in RAM (as indicated in the above table). The TOE clears these keys (i.e., plaintext keys in volatile storage), by destroying the reference to the key and requesting a garbage collection action. Additionally, the TOE is designed to zeroize secret and private keys when they are no longer required by the

TOE as detailed in **Table 6-3 Cryptographic Keys**. Note that zeroization occurs as follows: 1) when deleted from disk, the previous value is overwritten once with zeroes; 2) when added or changed on disk, any old value is overwritten completely with the new value; and, 3) the zeroization of values in RAM is achieved by overwriting once with zeroes.

- NDcPP22e:FCS_COP.1/DataEncryption: The TOE performs AES encryption and decryption. Refer to CAVP Tables 6-1 and 6-2 for the specifics regarding this cryptographic function and relevant CAVP certificates for the TOE cryptographic libraries.
- NDcPP22e:FCS_COP.1/Hash: The TOE performs cryptographic hashing using the algorithms shown in Tables 6-1, 6-2 and 6-3. Refer to these tables above for the specifics regarding this cryptographic function and relevant CAVP certificates.
- NDcPP22e:FCS_COP.1/KeyedHash: The TOE performs keyed-hashing using the algorithms shown in Tables 6-1 and 6-2. Refer to these tables above for the specifics regarding this cryptographic function and relevant CAVP certificates.
- NDcPP22e:FCS_COP.1/SigGen: The TOE performs cryptographic signature generation and verification using 2048-bit RSA schemes per FIPS 186-4 as shown in Tables 6-1, 6-2 and 6-3.
- NDcPP22e:FCS_HTTPS_EXT.1: An HTTPS/TLS connection is available which presents Web GUI and Rest API administrative interfaces. The TOE implements HTTPS per RFC 2818 with TLSv1.2. A connection can be established only if the peer initiates the connection.
- NDcPP22e:FCS_RBG_EXT.1: The TOE includes the IBM CTR_DRBG (AES256). This DRBG is used for all keys generated in support of cryptographic operations for TLS and SSH. The DRBG is seeded by drawing 256-bits from the Linux Kernel RNG which is augmented by noise from the JitterEntropy software noise source. The TOE also includes the IBMFIPSJCE HASH_DRBG which is seeded by drawing random data from the JitterEntropy software noise source with at least 256-bits of entropy.
- NDcPP22e:FCS_SSHS_EXT.1: The TOE supports SSHv2 as defined by RFCs 4251, 4252, 4253, 4254, 4256, 4344, 5647, 5656, 6187, 6668, 8268, and 8308 section 3.1, 8332. The TOE supports SSHv2 with AES (CBC) 128 or 256 bit ciphers, in conjunction with HMAC-SHA2-256, and HMAC-SHA2-512. The TOE allow use of ssh-rsa, rsa-sha2-256, and rsa-sha2-512 for authentication of the TOE to the client. The TOE supports the following key exchange methods: diffie-hellman-group14-sha1, and diffie-hellman-group14-sha256. While other ciphers and hashes are implemented in the product, they are disabled while the TOE is operating in FIPS mode. The TOE also supports RSA public-key authentication for users connecting to the TOE with a public-key.

  The SSHv2 authentication timeout period is 120 seconds allowing clients to retry only 5 times; both public-key and password based authentication can be configured. Packets are limited to 262127 bytes and are dropped if they exceed this limit. Whenever the timeout period or authentication retry limit is reached, the TOE closes the applicable TCP connection and releases the SSH session resources. As SSH packets are being received, the TOE uses a buffer to build all packet information. Once complete, the packet is checked to ensure it can be appropriately decrypted. However, if it is not complete when the buffer becomes full (262127 bytes) the packet will be dropped. An SSH session is rekeyed by the TOE after being established one hour or after transmitting 1Gb of traffic whichever comes first. A client initiated rekey, resets both the time and traffic counters.

- NDcPP22e:FCS_TLSC_EXT.1/2: The TOE provides TLS protected export of audit/event data with mutual authentication using X.509v3 certificates. When a TLS server requests TLS authentication via certificates, the TOE is capable of providing an X509v3 certificate.

  The TOE includes support for TLSv1.2 only. No older versions of TLS, and no version of SSL are supported. The TOE supports the ciphersuites identified above. When validating a certificate, the TOE ensures the distinguished name (DN) or Subject Alternate Name (SAN) fields in the certificate match the peer identifier DNS name. The TOE supports the use of wildcard values within a certificate’s DN or SAN field. The TOE does not support certificate pinning nor does it support an IP address as a reference identifier.

- NDcPP22e:FCS_TLSS_EXT.1: An HTTPS/TLS connection is available which presents a Web GUI administrative interface. Thus, the TOE acts as a TLS server supporting TLSv1.2 only. No older versions of TLS and no version of SSL are supported. The TOE supports the ciphersuites identified above.

  TLS v1.2 is supported with AES (CBC or GCM) 128 bit ciphers, in conjunction with SHA-256 and RSA. The TOE utilizes RSA with key sizes of 2048-bits for authentication and supports ECDHE key exchanges. The TOE support session resumption based on session tickets (adhering to the structural format from section 4 of RFC 5077). The session tickets are encrypted using AES CBC mode with a 128-bit key.

6.3 Identification and authentication

The TOE offers administrative operations which are accessible via the CLI (command line interface) or HTTPS/TLS protected interface only after a successful authentication.

The CLI is a text based interface which can be accessed from a directly connected terminal or via a remote terminal using SSH. Administrators are authenticated through the SSH CLI using either a username and password, or a public key. Users of the console and HTTPS/TLS interfaces are also authenticated using a username and password.

The Identification and authentication function satisfies the following security functional requirements:

- NDcPP22e:FIA_AFL.1: The TOE allows an administrator to configure the number of failed authentication attempts that trigger the locking of a user account. The limit of failed authentication attempts (a.k.a., the Max Login Failure limit) can be set to between 1 and 2,147,483,647. When the Max Login Failure limit is reached, the user account is locked. This prevents successful login of that account using all remote administration methods until the configured time limit has passed from the last failed authentication attempt. The configured lockout time is specified by an administrator and can be set to between 1 minute and 2,147,483,647 minutes. Admin accounts are never locked out from using the local console.
- NDcPP22e:FIA_PMG_EXT.1: The TOE supports user management capabilities which allow the administrator to configure a minimum length password between 15 and 255 characters (including a length of 15 characters). The TOE also supports the use of uppercase letters, lowercase letters, numeric values and special characters for passwords. The special characters permitted to be used in passwords including "!", "@", "#", "$", %", "^", "&", "*", "(" and ")".
- NDcPP22e:FIA_UAU.7: The TOE obscures passwords when entered by administrators.
- NDcPP22e:FIA_UAU_EXT.2: The TOE authenticates administrators using a local password-based mechanism or using a locally stored, public-key based authentication mechanism. The password-based mechanism is available for use through the local console CLI, the remote SSH CLI, the TLS-protected WebUI and the TLS-protected RestAPI. The public-key authentication method is supported only for SSH connections.
- NDcPP22e:FIA_UIA_EXT.1: The TOE does not offer any services through the administrative interfaces prior to authenticating the connecting user other than the display of a TOE warning banner. In order for an administrative user to access the TOE (i.e., to perform any functions except to see a configured login banner or to access network or SAN services), a user account including a user name and password must be created for the user. A successful logon occurs when the user presents to the TOE a valid administrative user name along with either the correct password or verification of a public-key.
- NDcPP22e:FIA_X509_EXT.1/Rev: The TOE supports the use of CRLs to determine the revocation status of certificates. Certificates are validated as part of the authentication process when they are presented to the TOE and when they are loaded into the TOE. The following fields are verified as appropriate:
  - Expiration – certificate cannot be expired.
  - Common Name - Needs to be FQN device name or IP. Wildcards are allowed only for 1 level of sub-domain and not allowed for the main domain. URI is not supported as an identifier type.
  - CA Field – must be true if CA certificate.

  - Key Usage - Need to have "Certificate Sign" in case of CA certificates and "Digital Signature" in case of identity certificates.
  - X509v3 Extended Key Usage - Need to rightly indicate whether it is for use as “server” certificate or “client” certificate. If incorrect, connection is not allowed.
  - X509v3 CRL Distribution Points- Certificate must be valid per a current CRL.
  - Subject Alternative Method - Not a mandatory attribute. If present, the values stored in this will take priority over the CN in Subject attribute.
  - Basics Constraints - Attribute must be present and must have CA Field.

- NDcPP22e:FIA_X509_EXT.2: The TOE uses X509v3 certificates for TLS communications. Certificates are validated as part of the authentication process when they are presented to the TOE and when they are loaded into the TOE. Administrators must install a trusted root CA certificate. The TOE must also install its own certificate which it will present to the syslog server during TLS negotiations. The certificate presented by a syslog peer and the certificate imported into the TOE must chain to this root CA certificate.

  If the TOE cannot establish a connected with a revocation server to check the status of a certificate in question, the TOE does not accept the certificate as valid.

- NDcPP22e:FIA_X509_EXT.3: An administrator can issue commands to generate a RSA CSR on the TOE. The administrator is asked for information to complete the CSR, including the following:
  - Country Name (2 letter code) [XX]:
  - State or Province Name (full name) []:
  - Locality Name (e.g., city) [Default City]:
  - Organization Name (e.g., company) [Default Company Ltd]:
  - Organizational Unit Name (e.g., section) []:
  - Common Name (e.g., your name or your server’s hostname) []:
  - Email Address []:

6.4 Security management

The TOE provides the ability for one role, “security administrators”, to securely configure and manage the TOE. Administrators can connect to the Command Line Interface (CLI) via either a remote SSH connection or using a locally connected terminal. The CLI is used for initial configuration, creating CSR requests, performing updates, reviewing local audit logs, and remediation of AIDE integrity check discrepancies. Administrators perform administrative tasks using an HTTPS/TLS protected communication channel offering a Web-based GUI.

Once authenticated (none of these functions is available to any user before being identified and authenticated), administrators have access to the following security 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 behavior (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);
- Ability to manage the cryptographic keys;
- Ability to configure the cryptographic functionality;
- Ability to configure thresholds for SSH rekeying;
- Ability to configure the reference identifier for the peer;
- Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;
- Ability to import X509v3 certificates to the TOE's trust store;
- Ability to set the time which is used for time-stamp.

The Security management function satisfies the following security functional requirements:

- NDcPP22e:FMT_MOF.1/ManualUpdate: Only administrators can initiate a software update through the command line interface only.
- NDcPP22e:FMT_MTD.1/CoreData: Only administrators can login to the TOE via its command line or Web GUI interfaces.
- NDcPP22e:FMT_MTD.1/CryptoKeys: Only administrators can import certificates and configure cryptography on the TOE. An administrator may generate and delete SSH host keys as well as keys associated with a CSR. The administrator may also import keys associated with CA certificates, as well as identity certificates that are used by the TOE.
- NDcPP22e:FMT_SMF.1: The TOE offers the ability to configure a login banner and an inactivity timeout value that can be used for session termination or session locking. The TOE also provides mechanisms to support an administrator's ability to verify and install TOE updates; to configure audit behavior; to set the time, and to configure cryptographic functionality. Specific functions are listed above.
- NDcPP22e:FMT_SMR.2: The TOE supports only a Security Administrator role. The TOE supports local administration via a CLI or a locally attached network device. Remote administration is provided through a CLI protected by SSH or through a web GUI protected by HTTPS/TLS. The majority of administrative functionality is available only through the web GUI.

6.5 Protection of the TSF

The TOE includes several mechanisms to support self-protection, including protection of sensitive data, accurate time, self-testing, and trusted updates.

The administrator can query the currently installed TOE version using commands on the CLI or via screens provided through the WebUI. The mechanism to support verification of TOE updates utilizes an RSA 2048-bit key. The IBM public key is installed as part of the TOE installation process. An update is delivered as an executable with an embedded payload. The embedded payload is a signed package. When executed, the payload is verified and if valid it is extracted. Upon failure of the signature check the extracted payload is deleted; upon success there is an indicator that installation can proceed.

The Protection of the TSF function satisfies the following security functional requirements:

- NDcPP22e:FPT_APW_EXT.1: Administrator passwords are stored SHA-512 hashed and salted. No interface is offered by the TOE to present cleartext passwords.
- NDcPP22e:FPT_SKP_EXT.1: Pre-shared keys, symmetric keys, and private keys stored by the TOE are AES encrypted and never presented back to administrators as plaintext. The AES key protecting other keys is stored in a password protected PKCS12 file and is never presented or displayed in any tools or UI. The password to access the PCKS12 file is used by the TOE to decrypt the AES key, which allows all other keys to become accessible to the TOE software.
- NDcPP22e:FPT_STM_EXT.1: The TOE uses the system time obtained from the hardware for use in audit records, session timeouts, scheduling integrity validation and during certificate validation. The hardware-based time value is considered reliable because it can be set and modified only by an administrator.
- NDcPP22e:FPT_TST_EXT.1: The TOE runs a suite of cryptographic self-tests during power-on and performs TSF integrity checks daily. These tests include cryptographic known answer tests exercising the algorithms shown in **Table 6-1 CAVP Certificates for OpenSSL library** and **Table 6-2 CAVP Certificates for IBMFIPSJCE library**. A failure of any of the power-up self-tests panics the module. The only recovery is to reboot. If a reboot cannot solve the problem, the system may be compromised, and QRadar must be reinstalled on the system.

  For TSF integrity checking, the TOE uses Advanced Intrusion Detection Environment (AIDE) tool to verify the integrity of TOE software. AIDE (https://aide.github.io/) is a utility that creates a database of files on the system, and then uses that database to ensure file integrity and detect system intrusions. After installation and configuration the administrator creates an AIDE baseline database and schedules the integrity check to run at configured intervals using cron. In the case where the AIDE tool identifies differences in the database the Administrator will review the identified system changes. If all changes are approved the administrator will commit the new database, otherwise the integrity of the device will be restored through a factory re-installation.

These tests together ensure that the TOE continues to operate correctly because they test cryptographic operations used by the TOE and verify all TOE software is unchanged.

In the event a self-test fails, the administrator is expected to make a system image of the appliance for later investigation, and then reinstall the system. Since the TOE provides services to other network devices, availability is an important consideration in its design. To manage the time needed during startup, the AIDE tools is not run during TOE initialization, but rather is scheduled on a daily basis. This allows the TOE to return to providing network services quicker and to perform the TSF integrity checking more frequently. Guidance suggests that the AIDE tool be scheduled on a daily basis, however, administrators can schedule it more frequently if desired.

- NDcPP22e:FPT_TUD_EXT.1: IBM signs product updates using a 2048-bit RSA key that is installed on the TOE. The SHA-256 signature on an update is verified prior to installing the update (as described above). Administrators must manually obtain updates and must take explicit actions to install an update (the TOE does not automatically update itself). The TOE UI presents the currently running version upon request.

6.6 TOE access

The TOE access function satisfies the following security functional requirements:

- NDcPP22e:FTA_SSL.3: For remote sessions, the TOE provides an inactivity timeout mechanism that is configurable by an administrator. The timeout mechanism applies to the HTTPS-protected Web GUI and to the SSH-protected CLI interface.
- NDcPP22e:FTA_SSL.4: All administrative users can issue a logout from a remote administrative session using either the SSH, or the Web GUI interface. The RESTAPI interface is not an interactive interface, but rather every RESTAPI invocation is atomic and each is authenticated. Administrators can also logout from a local console session.
- NDcPP22e:FTA_SSL_EXT.1: The TOE terminates local sessions that have been inactive for an administrator-configured period of time. After termination, administrative authentication is required to access any of the administrative functionality of the TOE.
- NDcPP22e:FTA_TAB.1: The TOE presents a warning banner to the user during login actions prior to completing authentication on the Web GUI, the SSH interface and the local console.

6.7 Trusted path/channels

The Trusted path/channels function satisfies the following security functional requirements:

- NDcPP22e:FTP_ITC.1: Communication with an external audit server (i.e., outbound) will utilize mutually authenticated TLS. The external audit server would be expected to accept TLS communication initiated by the TOE.
- NDcPP22e:FTP_TRP.1: The TOE provides multiple methods of remote administration. A command line interface is available remotely via an SSH protected channel. Additionally, an HTTPS/TLS connection is available which presents a Web GUI administrative interface and the RESTAPI interface. Administrators must initiate the connection to the TOE from a remote network entity.