NETSCOUT

TestStream Management Software v5.3.0 on nGenius 3900 Series Switches

Security Target

Version 1.3

April 2023

Document prepared by

www.lightshipsec.com

Document History

Version	Date	Description
1.0	16 Feb 2023	Release for Certification
1.1	1 Mar 2023	Certification updates
1.2	31 Mar 2023	Update Guidance document version references
1.3	14 Apr 2023	Address CB Comments

Table of Contents

List of Tables

1 Introduction

1.1 Overview

1 This Security Target (ST) defines the NETSCOUT TestStream Management Software v5.3.0 on nGenius 3900 Series Switches Target of Evaluation (TOE) for the purposes of Common Criteria (CC) evaluation.

2 NETSCOUT's TestStream Management Software provides both web and CLI interfaces for management of intelligent, reconfigurable layer 1 fabrics. Drag-and-drop functionality allows for ease of management for layer 1 switching and layer 2-4 functions.

1.2 Identification

Table 1: Evaluation identifiers

Target of Evaluation	TestStream Management Software v5.3.0 on nGenius 3900 Series Switches Build: 5.3.0.54
Security Target	TestStream Management Software v5.3.0 on nGenius 3900 Series Switches Security Target, v1.3

1.3 Conformance Claims

3 This ST supports the following conformance claims:

a) CC Version 3.1 Revision 5

b) CC Part 2 Extended

c) CC Part 3 Conformant

d) collaborative Protection Profile for Network Devices, v2.2e

e) NIAP Technical Decisions per Table 2

Table 2: NIAP Technical Decisions

TD #	Name	Applicability	Exclusion Rationale
TD0527	Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	
TD0528	NIT Technical Decision for Missing EA's for FCS_NTP_EXT.1.4	No	FCS_NTP_EXT.1 not claimed
TD0536	NIT Technical Decision for Update Verification Inconsistency	Yes	
TD0537	NIT Technical Decision for Incorrect Reference to FCS_TLSC_EXT.2.3	No	FCS_TLSC_EXT.2 not claimed
TD0538	NIT Technical Decision for Outdated Link to Allowed-with List	No	Only applicable to Protection Profile

TD #	Name	Applicability	Exclusion Rationale
TD0546	NIT Technical Decision for DTLS – clarification of Application Note 63	No	FCS_DTLS not claimed
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes	
TD0555	NIT Technical Decision for RFC Reference incorrect in TLSS Test	Yes	
TD0556	NIT Technical Decision for RFC 5077 question	Yes	
TD0563	NiT Technical Decision for Clarification of Audit Date Information	Yes	
TD0564	NiT Technical Decision for Vulnerability Analysis Search Criteria	Yes	
TD0569	NiT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	No	FCS_DTLSS not claimed
TD0570	NiT Technical Decision for Clarification about FIA_AFL.1	Yes	
TD0571	NiT Technical Decision for Guidance on How to Handle FIA_AFL.1	Yes	
TD0572	NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes	
TD0580	NiT Technical Decision for Clarification about use of DH14 in NDcPPv2.2e	Yes	
TD0581	NiT Technical Decision for Elliptic Curve-based key establishment and NIST SP 800-56A rev3	Yes	
TD0591	NIT Technical Decision for Virtual TOEs and hypervisors	No	TOE is not a virtual TOE
TD0592	NIT Technical Decision for Local Storage of Audit Records	Yes	
TD0631	NIT Technical Decision for Clarification of public key authentication for SSH Server	Yes	
TD0632	NIT Technical Decision for Consistency with Time Data for vNDs	Yes	
TD0633	NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	No	FCS_IPSEC_EXT.1 not claimed

TD #	Name	Applicability	Exclusion Rationale
TD0634	NIT Technical Decision for Clarification required for testing IPv6	No	FCS_TLSC_EXT.1 or FCS_DTLSC_EXT.1 not claimed
TD0635	NIT Technical Decision for TLS Server and Key Agreement Parameters	Yes	
TD0636	NIT Technical Decision for Clarification of Public Key User Authentication for SSH	Yes	
TD0638	NIT Technical Decision for Key Pair Generation for Authentication	Yes	
TD0639	NIT Technical Decision for Clarification for NTP MAC Keys	No	FCS_NTP_EXT.1 not claimed
TD0670	NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	No	FCS_TLSC_EXT.1 or FCS_TLSC_EXT.2 not claimed.

1.4 Terminology

Table 3: Terminology

Term	Definition
CC	Common Criteria
CC Version 3.1 Revision 5	Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model, April 2017, Version 3.1 Revision 5, CCMB-2017-04-001
CC Part 2	Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components, April 2017, Version 3.1 Revision 5, CCMB-2017-04-002
CC Part 3	Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Components, April 2017, Version 3.1 Revision 5, CCMB-2017-04-003
NDcPP	collaborative Protection Profile for Network Devices
PP	Protection Profile
TD	Technical Decision
TOE	Target of Evaluation
TSF	TOE Security Functionality
PFS	Packet Flow Switch (NETSCOUT)

Term	Definition
CLI	Command Line Interface
TS	TestStream (NETSCOUT Software)

2 TOE Description

2.1 Type

4 The TOE is a network device that provides management features to modify and monitor the layer 1 fabric with applications in test labs, customer support labs, and other environments.

2.2 Usage

2.2.1 Deployment

5 The TOE is deployed within networks that provide connectivity to environments described in section 2.1. The TOE is deployed locally on the switch in which it resides.

2.2.2 Interfaces

6 The TOE interfaces within the scope of evaluation are shown in Figure 1.

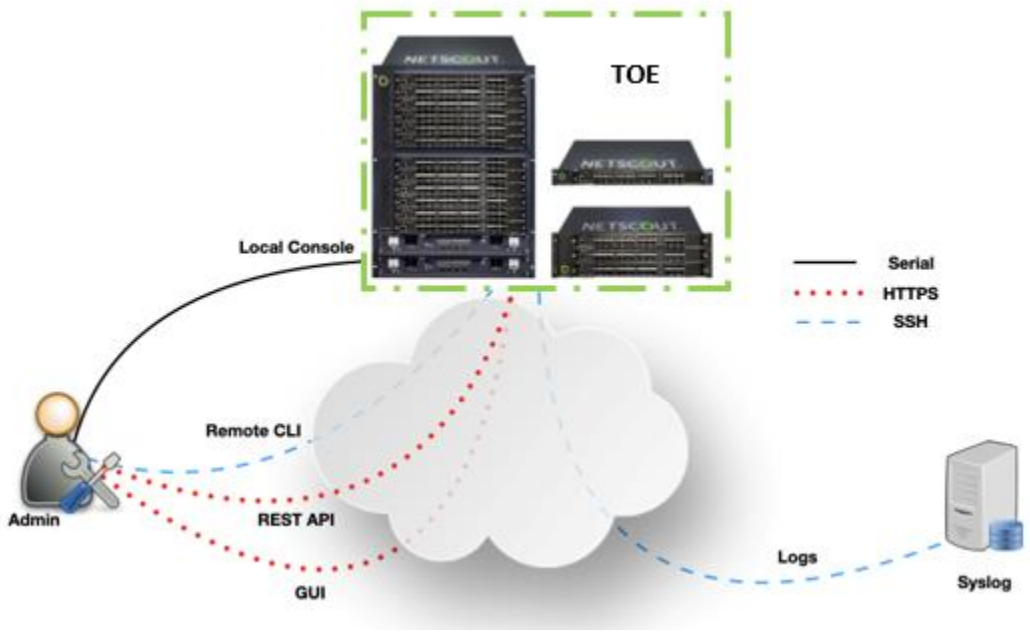

Figure 1: TOE interfaces

7 The logical TOE interfaces are as follows:

a) **TestStream Management GUI.** A Java applet that leverages the stunnel service to provide the TestStream Management GUI. The applet is launched from a browser over HTTPS.

b) **Remote CLI (SSHv2).** Administrative CLI via SSH for operating system functions, initial TOE configuration and maintenance operations, in addition to the administrative TestStream CLI.

c) **REST API (HTTPS).** REST API provided by the apache service for programmatic administration via HTTPS.

d) **Local Console (Serial).** Provides local access for initial TOE configuration and maintenance operations, and administrative TestStream CLI.

e) **Syslog (SSHv2).** The TOE sends audit events to syslog over SSH.

2.3 Security Functions / Logical Scope

8 The TOE provides the following security functions:

a) **Protected Communications.** The TOE protects the integrity and confidentiality of communications as noted in section 2.2 above.

b) **Secure Administration.** The TOE enables secure management of its security functions, including:

i) Administrator authentication with passwords

ii) Configurable password policies

iii) Role Based Access Control

iv) Access banners

v) Management of critical security functions and data

vi) Protection of cryptographic keys and passwords

c) **Trusted Update.** The TOE ensures the authenticity and integrity of software updates through published hashes.

d) **System Monitoring.** The TOE generates logs of security relevant events. The TOE stores logs locally and is capable of sending log events to a remote audit server.

e) **Self-Test.** The TOE performs a suite of self-tests to ensure the correct operation and enforcement of its security functions.

f) **Cryptographic Operations.** The TOE implements a cryptographic module. Relevant Cryptographic Algorithm Validation Program (CAVP) certificates are shown in Table 4.

Table 4: CAVP Certificates

Operation	Capability	Key Size	Certificate
Encryption/ Decryption	AES-CBC, AES-CTR, AES-GCM (CTR_DRBG)	128, 256	C2144
Signature Generation	ECDSA Key Gen (186-4) ECDSA Sig Gen / Sig Ver (186-4)	P-256, P-384, P-521	

Operation	Capability	Key Size	Certificate
Hash	SHA-1, SHA-256, SHA-384, SHA-512	160, 256, 384, 512	
Keyed Hash	HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512	256, 384, 512	
DRBG	CTR_DRBG (AES)	n/a	
Key Agreement	KAS-ECC-SSC	P-256/P-384/P-521	A1882

2.3.1 Functions not included in the TOE Evaluation

9 The evaluation is limited to those security functions identified in section 2.3 above.

10 The management features to modify and monitor the layer 1 fabric with applications in test labs, customer support labs, and other environments are outside of the scope of the evaluated security functions.

11 Use of NTP protocol is not addressed by this evaluation.

12 Use of Syslog over TLS is not addressed by this evaluation.

2.4 Physical Scope

13 The physical boundary of the TOE includes the TestStream Management Software operating on the hardware shown in Table 5.

14 NETSCOUT 3900 series switches includes a switch chassis and blade combination. The switch chassis component houses the blades and provides power. The blades run the TOE software and provide management and networking interfaces and services. In a multi-blade configuration, only one blade can assume the role of the active controller. The active controller manages all other blades in the switch.

15 The TOE is delivered via commercial courier.

Table 5: TOE models

Model	Manufacturer	Processor
3901R Switch (1 blade capacity)	NETSCOUT	(Chassis)
3903 Switch (3 blade capacity)	NETSCOUT	(Chassis)
3912 Switch (12 blade capacity)	NETSCOUT	(Chassis)
S-Blade Pro (24 QSFP+ ports)	NETSCOUT	NXP QorIQ® P4081 (Timesys Linux 3.8.13)

Model	Manufacturer	Processor
S-Blade 64 (32 SFP+ Ports and 8 QSFP+ ports)	NETSCOUT	NXP QorIQ® P4081 (Timesys Linux 3.8.13)

2.4.1 Guidance Documents

16 The TOE includes the following guidance documents in PDF file format which are available through the NETSCOUT Mastercare web portal:

a) NETSCOUT TestStream Management Software v5.3.0 on nGenius 3900 Series Switches Security Target v1.3

b) NETSCOUT TestStream Management Software 5.3.0 Administrator Guide 733-1696 Rev. A

c) NETSCOUT TestStream Management Software v5.3.0 on nGenius 3900 Series Switches Common Criteria Guide v1.3

d) NETSCOUT TestStream Management Software v5.3.0 Common Criteria Addendum Rev 4.10

Note: A myNetscout Account is required to access these documents via the Mastercare website.

2.4.2 Non-TOE Components

17 The TOE operates with the following components in the environment:

a) **Syslog Server.** The TOE is capable of sending audit events to a Syslog server.

3 Security Problem Definition

18 The Security Problem Definition is reproduced from the NDcPP.

3.1 Threats

Table 6: Threats

Identifier	Description
T.UNAUTHORIZED_ ADMINISTRATOR_ ACCESS	Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_ CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.
T.UNTRUSTED_ COMMUNICATION_ CHANNELS	Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself.
T.WEAK_ AUTHENTICATION_ ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.
T.UPDATE_ COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validate d using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and

Identifier	Description
	the Administrator would have no knowledge that the device has been compromised.
T.SECURITY_ FUNCTIONALITY_ COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker.
T.PASSWORD_ CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic and may allow them to take advantage of any trust relationships with other Network Devices.
T.SECURITY_ FUNCTIONALITY_ FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

3.2 Assumptions

Table 7: Assumptions

Identifier	Description
A.PHYSICAL_ PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorize d entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_ FUNCTIONALITY *(Modified by TD0591)*	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.
A.NO_THRU_ TRAFFIC_ PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall).
A.TRUSTED_ ADMINISTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).

Identifier	Description
A.REGULAR_ UPDATES	The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_ CREDENTIALS_ SECURE	The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.RESIDUAL_ INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

3.3 Organizational Security Policies

Table 8: Organizational Security Policies

Identifier	Description
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

4 Security Objectives

19 The security objectives are reproduced from section 5 of the NDcPP.

Table 9: Security Objectives for the Operational Environment

Identifier	Description
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_ PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. 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.

Identifier	Description
OE.TRUSTED_ADMIN	Security 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.ADMIN_ CREDENTIALS_ SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.RESIDUAL_ INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.

5 Security Requirements

5.1 Conventions

20 This document uses the following font conventions to identify the operations defined by the CC:

a) **Assignment.** Indicated with italicized text.

b) **Refinement.** Indicated with bold text and strikethroughs.

c) **Selection.** Indicated with underlined text.

d) **Assignment within a Selection:** Indicated with italicized and underlined text.

e) **Iteration.** Indicated by adding a string starting with "/" (e.g. "FCS_COP.1/Hash").

21 **Note:** Operations performed within the Security Target are denoted within brackets []. Operations shown without brackets are reproduced from the NDcPP.

5.2 Extended Components Definition

22 The Extended Components shown in Table 10 are defined in Appendix C of the NDcPP.

Table 10: Extended Components

Requirement	Title	Applicable TDs
FAU_STG_EXT.1	Protected Audit Event Storage	
FCS_HTTPS_EXT.1	HTTPS Protocol	
FCS_RBG_EXT.1	Random Bit Generation	
FCS_SSHC_EXT.1	SSH Client Protocol	
FCS_SSHS_EXT.1	SSH Server Protocol	TD0631
FCS_TLSS_EXT.1	TLS Server Protocol Without Mutual Authentication	TD0635
FIA_PMG_EXT.1	Password Management	
FIA_UIA_EXT.1	User Identification and Authentication	
FIA_UAU_EXT.2	Password-based Authentication Mechanism	
FIA_X509_EXT.1	X.509 Certificate Validation	TD0527
FIA_X509_EXT.2	X.509 Certification Authentication	
FIA_X509_EXT.3	X.509 Certificate Requests	
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)	
FPT_APW_EXT.1	Protection of Administrator Passwords	
FPT_TST_EXT.1	TSF Testing	
FPT_TUD_EXT.1	Trusted Update	
FPT_TUD_EXT.2	Trusted Update Based on Certificates	
FPT_STM_EXT.1	Reliable Time Stamps	
FTA_SSL_EXT.1	TSF-initiated Session Locking	

5.3 Functional Requirements

Table 11: Summary of SFRs

Requirement	Title
FAU_GEN.1	Audit Data Generation
FAU_GEN.2	User Identity Association

Requirement	Title
FAU_STG_EXT.1	Protected Audit Event Storage
FCS_CKM.1	Cryptographic Key Generation
FCS_CKM.2	Cryptographic Key Establishment
FCS_CKM.4	Cryptographic Key Destruction
FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
FCS_HTTPS_EXT.1	HTTPS Protocol
FCS_RBG_EXT.1	Random Bit Generation
FCS_SSHS_EXT.1	SSH Server Protocol
FCS_SSHC_EXT.1	SSH Client Protocol
FCS_TLSS_EXT.1	TLS Server Protocol
FIA_AFL.1	Authentication Failure Management
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU_EXT.2	Password-based Authentication Mechanism
FIA_UAU.7	Protected Authentication Feedback
FIA_X509_EXT.1/Rev	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FIA_X509_EXT.3	X.509 Certificate Requests
FMT_MOF.1/ManualUpdate	Management of Security Functions by Behaviour
FMT_MOF.1/Functions	Management of Security Functions Behaviour
FMT_MTD.1/CoreData	Management of TSF Data
FMT_MTD.1/CryptoKeys	Management of TSF Data

Requirement	Title
FMT_SMF.1	Specification of Management Functions
FMT_SMR.2	Restrictions on Security Roles
FPT_SKP_EXT.1	Protection of TSF Data (For reading of all pre-shared symmetric and private keys)
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_TST_EXT.1	TSF Testing (Extended)
FPT_TUD_EXT.1	Trusted Update
FPT_STM_EXT.1	Reliable Time Stamps
FTA_SSL_EXT.1	TSF-initiated Session Locking
FTA_SSL.3	TSF-initiated Termination
FTA_SSL.4	User-initiated Termination
FTA_TAB.1	Default TOE Access Banners
FTP_ITC.1	Inter-TSF Trusted Channel
FTP_TRP.1/Admin	Trusted Path

5.3.1 Security Audit (FAU)

FAU_GEN.1 Audit Data Generation

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

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

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

c) All administrative actions 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 2~~ Table 12****.*

Table 12: Audit Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS session.	Reason for failure.
FCS_RBG_EXT.1	None.	None.
FCS_SSHC_EXT.1	Failure to establish an SSH session.	Reason for failure.
FCS_SSHS_EXT.1	Failure to establish an SSH session.	Reason for failure.
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
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).

Requirement	Auditable Events	Additional Audit Record Contents
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store.	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MOF.1/Functions	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	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 "lock the session" is selected)	Any attempts at unlocking of an interactive session.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FTA_SSL_EXT.1 (if "terminate the session" is selected)	The termination of a local 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.

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 2~~ Table 12***.

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.

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.

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]

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

5.3.2 Cryptographic Support (FCS)

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: [

- 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;

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

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 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";

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

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* [single overwrite consisting of [zeroes]],
- *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 [*10-pass*] overwrite consisting of [zeroes]];

that meets the following: *No Standard*.

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, GCM]

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, GCM as specified in ISO 19772]

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

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

- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [*256 bits or greater]*

]

that meet the following: [

- 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- 256, HMAC-SHA-384, HMAC-SHA-512] and cryptographic key sizes [*256, 384, 512*] **and message digest sizes [256, 384, 512] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".*

FCS_HTTPS_EXT.1 HTTPS Protocol

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

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3 If a peer certificate is presented, the TSF shall [not require client authentication] if the peer certificate is deemed invalid.

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 [CTR_DRBG (AES)]

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 18031:2011 Table C.1 “Security Strength Table for Hash Functions”, of the keys and hashes that it will generate.

FCS_SSHC_EXT.1 SSH Client Protocol

FCS_SSHC_EXT.1.1 The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [4344, 5656, 6668, 8268, 8308 section 3.1].

FCS_SSHC_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, [no other method].

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

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

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

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

FCS_SSHC_EXT.1.7 The TSF shall ensure that [ecdh-sha2-nistp256] and [diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, ecdh-sha2-nistp384, ecdh-sha2-nistp521] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHC_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.

FCS_SSHC_EXT.1.9 The TSF shall ensure that the SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key and [no other methods] as described in RFC 4251 section 4.1.

FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [4256, 4344, 5656, 6668, 8268, 8308 section 3.1]

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]

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*256 kilo*]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-ctr, aes256-ctr].

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

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

FCS_SSHS_EXT.1.7 The TSF shall ensure that [ecdh-sha2-nistp256, diffie-hellman-group14-sha1] and [diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, ecdh-sha2-nistp384, ecdh-sha2-nistp521] are the only allowed key exchange methods used for the SSH protocol.

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

FCS_TLSS_EXT.1 TLS Server Protocol Without Mutual Authentication

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_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
-] and no other ciphersuites.

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].

FCS_TLSS_EXT.1.3 The TSF shall perform key establishment for TLS using [ECDHE curves [secp256r1, secp384r1, secp521r1] and no other curves]].

FCS_TLSS_EXT.1.4 The TSF shall support [session resumption based on session tickets according to RFC 5077].

5.3.3 Identification and Authentication (FIA)

FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [*0 to 10*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password.*

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met, the TSF shall [prevent the offending Administrator from successfully establishing a remote session using any authentication method that involves a password until [*an Administrator conducts a password reset operation via the local console interface*] is taken by an Administrator]

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: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, *[“+”, “=”, “_”, “.”, “-”]*];

b) Minimum password length shall be configurable to between [*1*] and [*30*] characters

FIA_UIA_EXT.1 User Identification and Authentication

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

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

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

FIA_UAU_EXT.2 Password-based Authentication Mechanism

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

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**.

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 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 [the Online Certificate Status Protocol (OCSP) as specified in RFC 6960]
- 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.

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 [HTTPS, TLS] and [no additional uses]

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].

FIA_X509_EXT.3 X.509 Certificate Requests

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

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

5.3.4 Security Management (FMT)

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.*

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] *to Security Administrators*.

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*.

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*

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* [hash comparison] *capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- [
    - Ability to configure audit behavior;
    - Ability to modify the behaviour of the transmission of audit data to an external IT entity;
    - Ability to manage the cryptographic keys;
    - Ability to re-enable an Administrator account;
    - Ability to set the time which is used for time-stamps;
    - Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;
    - Ability to import X.509v3 certificates to the TOE's trust store;

- Ability to manage the trusted public keys database;
- No other capabilities]

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.

5.3.5 Protection of the TSF (FPT)

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.

FPT_APW_EXT.1 Protection of Administrator Passwords

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

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

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: [

- *BIOS tests*
- *Cryptographic Algorithm Tests*
- *DRBG Tests*
- *Software Integrity Tests*]

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 [the most recently installed version of the TOE firmware/software]

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 [published hash] prior to installing those updates.

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]

5.3.6 TOE Access (FTA)

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.

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*.

FTA_SSL.4 User-initiated Termination

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

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.

5.3.7 Trusted path/channels (FTP)

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 [*audit server*]

FTP_TRP.1 /Admin Trusted Path

FTP_TRP.1.1/Admin	The TSF shall **be capable of using [SSH, TLS, and 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.**
FTP_TRP.1.2 /Admin	The TSF shall permit remote **Administrators** to initiate communication via the trusted path.
FTP_TRP.1.3 /Admin	The TSF shall require the use of the trusted path for *initial Administrator authentication and all remote administration actions.*

5.4 Assurance Requirements

23 The TOE security assurance requirements are summarized in Table 13.

Table 13: Assurance Requirements

Assurance Class	Assurance Components
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)

24 In accordance with section 7.1 of the NDcPP v2.2e, the following refinement is made to ASE:

a) **ASE_TSS.1.1C Refinement:** The TOE summary specification shall describe how the TOE meets each SFR. **In the case of entropy analysis, the TSS is used in conjunction with required supplementary information on Entropy.**

6 TOE Summary Specification

25 The following describes how the TOE fulfils each SFR included in section 5.3.

6.1 Security Audit

6.1.1 FAU_GEN.1

26 The TOE generates the audit records specified at Table 12.

27 The following information is logged as a result of the Security Administrator generating/importing or deleting cryptographic keys:

a) **Generate SSH keys.** Action and key reference.

b) **Generate CSR.** Action and key reference**.**

c) **Import Certificate.** Action and key reference.

d) **Import CA Certificate.** Action and key reference.

6.1.2 FAU_GEN.2

28 The TOE includes the user identity in audit events resulting from actions of identified users.

6.1.3 FAU_STG_EXT.1

29 The Security Administrator can configure the TOE to transmit generated audit data to an external IT entity including the transmission of logs to a Syslog server. Log events are sent in real-time. Logs are sent via SSH as described by FCS_SSHC_EXT.1.

30 The TOE consists of a single standalone component that stores audit data locally. By default, logs are stored locally and rotated as follows:

a) **/var/log log files.** Log files are rotated when a TestStream software update is installed in addition to an algorithm running every ten seconds which addresses the following: if the /var partition reaches 98% utilization or more, the TOE truncates system log files and removes rotated system log files. Otherwise, If the /var partition reaches 90% utilization or more, rotated log files are removed. If the /var partition reaches 50% utilization or more, auth.log and system log files are truncated, otherwise logrotate is run.

31 The amount of audit data that may be stored locally is dependent on the available disk space.

32 Only authorized administrators may view audit records and no capability to modify the audit records is provided.

6.2 Cryptographic Support

6.2.1 FCS_CKM.1

33 The TOE supports key generation for the following asymmetric schemes:

a) **ECC P-256/P-384/P-521.** Used in SSH and TLS.

6.2.2 FCS_CKM.2

34 The TOE supports the following key establishment schemes:

a) **ECC schemes.** Used in TLS and SSH ciphersuites with ECDH key exchange. TOE is both client and server.

Table 14: Key Agreement Mapping

SFR	Service	Key Agreement Schemes
FCS_TLSS_EXT.1	GUI / Administration REST API / Administration	ECC
FCS_SSHC_EXT.1	Audit Server	ECC
FCS_SSHS_EXT.1	CLI / Administration	ECC

6.2.3 FCS_CKM.4

35 Keys held in volatile memory are zeroized after use by overwriting the key storage area with zeroes. Keys held in persistent memory may be destroyed using a Command Line Interface (CLI) command to overwrite files containing keys. Table 16 shows the origin, storage location and destruction details for cryptographic keys and passwords. Unless otherwise stated, the keys are generated by the TOE.

6.2.4 FCS_COP.1/DataEncryption

36 The TOE provides symmetric encryption and decryption capabilities using 128 and 256 bit AES in CBC, CTR and GCM mode. AES is implemented in the following protocols: TLS and SSH.

37 The relevant NIST CAVP certificate numbers are listed Table 4.

6.2.5 FCS_COP.1/SigGen

38 The TOE provides cryptographic signature generation and verification services using:

a) ECDSA Signature Algorithm with NIST curves P-256, P-384, P-521.

39 These ECDSA signature verification services are used in the TLS protocols. Additionally, ECDSA signature verification is used for the SSH protocol.

40 The relevant NIST CAVP certificate numbers are listed in Table 4.

6.2.6 FCS_COP.1/Hash

41 The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512.

42 SHS is implemented in the following parts of the TSF:

a) TLS and SSH;

b) Hashing of passwords in non-volatile storage.

43 The relevant NIST CAVP certificate numbers are listed in Table 4.

6.2.7 FCS_COP.1/KeyedHash

44 The TOE provides keyed-hashing message authentication services using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512.

45 HMAC is implemented in the following protocols: TLS and SSH.

46 The characteristics of the HMACs used in the TOE are given in Table 15.

Table 15: HMAC Characteristics

Algorithm	Block Size	Key Size	Digest Size
HMAC-SHA-256	512 bits	256 bits	256 bits
HMAC-SHA-384	1024 bits	384 bits	384 bits
HMAC-SHA-512	1024 bits	512 bits	512 bits

47 The relevant NIST CAVP certificate numbers are listed in Table 4.

6.2.8 FCS_HTTPS_EXT.1

48 The TOE web GUI is a Java application downloaded via an HTTPS connection and uses the TLS implementation described by FCS_TLSS_EXT.1 for functional communication once installed. The TOE does not use HTTPS in a client capacity. The TOE's HTTPS protocol complies with RFC 2818.

49 RFC 2818 specifies HTTP over TLS. The majority of RFC 2818 is spent on discussing practices for validating endpoint identities and how connections must be setup and torn down. The TOE web GUI uses TLS over ports 60100 and 60101 which are designed to natively speak TLS: it does not attempt STARTTLS or similar multi-protocol negotiation which is described in section 2.3 of RFC 2818. The TLS server attempts to send closure Alerts prior to closing a connection in accordance with section 2.2.2 of RFC 2818.

6.2.9 FCS_RBG_EXT.1

50 The TOE contains a CTR_DRBG that is seeded from the software entropy source (Jitter RNG). Entropy from the noise source is extracted 256 bits at a time, conditioned and used to seed the DRBG with 256 bits of full entropy.

51 Additional detail is provided in the proprietary Entropy Description document.

6.2.10 FCS_SSHC_EXT.1

52 The TOE operates as an SSH client for the trusted channel with the Audit Server.

53 The TOE supports public key authentication using ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521 algorithms and will reject all other algorithms.

54 The TOE ensures that the SSH client authenticates the identity of the SSH server by using a local database that contains associations for the host name and corresponding public key.

55 The TOE examines the size of each received SSH packet. If the packet is greater than 256KB, it is automatically dropped.

56 The TOE utilises AES-CTR-128 and AES-CTR-256 for SSH encryption.

57 The TOE provides data integrity for SSH connections via HMAC-SHA2-256 and HMAC-SHA2-512.

58 The TOE supports ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521, diffie-hellman-group14-sha256, and diffie-hellman-group16-sha512 for SSH key exchanges.

59 The TOE will re-key SSH connections after 60 minutes or after an aggregate of 500MB of data has been exchanged (whichever occurs first).

6.2.11 FCS_SSHS_EXT.1

60 The TOE CLI is remotely accessed using the SSH implementation.

61 The TOE supports password-based or public key authentication using ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, and ecdsa-sha2-nistp521 algorithms and will reject all other algorithms. In the case of public keys, the TOE verifies the identity of the client using entries listed in the local authorized_keys file which identifies authorized hosts and maps them to their corresponding public key.

62 The TOE examines the size of each received SSH packet. If the packet is greater than 256KB, it is automatically dropped.

63 The TOE utilises AES-CTR-128 and AES-CTR-256 for SSH encryption.

64 The TOE provides data integrity for SSH connections via HMAC-SHA2-256 and HMAC-SHA2-512.

65 The TOE supports ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521, diffie-hellman-group14-sha1, diffie-hellman-group14-sha256, and diffie-hellman-group16-sha512 for SSH key exchanges.

66 The TOE will re-key SSH connections after 60 minutes of after an aggregate of 500MB of data has been exchanged (whichever occurs first).

6.2.12 FCS_TLSS_EXT.1

67 The TOE operates as a TLS server for HTTPS trusted paths.

68 The server only allows TLS protocol version 1.2. Any other protocol versions are rejected.

69 The TOE is restricted to the following ciphersuites by default:

a) TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289

b) TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289

c) TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289

d) TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

70 Ciphersuites are not user-configurable.

71 The TLS server performs key establishment for stunnel using ECDHE curves secp256r1, and key establishment for apache using ECDHE curves secp256r1, secp384r1, and secp521r1.

72 The TOE supports session resumption based on session tickets that are compliant with RFC5077. Session tickets are protected using the supported 128-bit and 256-bit AES encryption in CBC, CTR, and GCM modes as described in FCS_COP.1/DataEncryption.

6.3 Identification and Authentication

6.3.1 FIA_AFL.1

73 The TOE is capable of tracking authentication failures of remote administrators using interactive interfaces including the Web GUI and REST API.

74 When a user account has sequentially failed authentication after the configured number of attempts, the account will be locked until a Security Administrator resets the account password.

75 The administrator can configure the maximum number of failed attempts using the GUI.

76 The TestStream local console does not enforce the lockout mechanism for the built-in administrator.

6.3.2 FIA_PMG_EXT.1

77 The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters including but not limited to“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*” and the additional characters “+”, “=”, “_”, “.”, “-“.

78 The minimum password length is configurable to between 1 and 30 characters by the Security Administrator.

79 The maximum password length is 95 characters.

6.3.3 FIA_UIA_EXT.1

80 The TOE requires all users to be successfully identified and authenticated prior to gaining access to administrative functions. The TOE displays a warning banner prior to authentication as described by FTA_TAB.1

81 Administrative access to the TOE is facilitated through one of several interfaces:

a) **Local console (Serial).** Direct connection to the TOE appliance.

b) **Remote CLI (SSHv2)**. Remotely connecting via SSH protocol.

c) **TestStream Management GUI (TLS).** Remotely connecting to the TOE via TLS using the TestStream Management GUI Java applet.

d) **REST API (HTTPS).** Interactive administration using the REST API over HTTPS.

6.3.4 FIA_UAU_EXT.2

82 The TOE uses a local password-based authentication mechanism.

83 Interactive interfaces prompt the administrator for a username and password credential (including with each REST API session). Only after successful authentication with authorized credentials will the administrator be granted access to the TOE administrative functions. No access is permitted to the TOE unless an administrator is successfully identified and authenticated.

84 The authentication process is the same for interactive access occurring via direct console connection or remotely. At initial login, the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the password associated with the user account. If the logon is successful, the TOE will allow access to administrative functions according to the role in which

the administrator is assigned. The TOE does not provide verbose error messages in the event of a login failure.

6.3.5 FIA_UAU.7

85 For all authentication at the local CLI the TOE provides no feedback when the administrative password is entered so that the password is obscured.

6.3.6 FIA_X509_EXT.1/Rev

86 The TOE performs X.509 certificate validation at the following points:

a) When certificates are loaded into the TOE, such as when importing CAs, certificate responses and other device-level certificates (such as the web server certificate presented by the TOE TLS web GUI).

87 In all scenarios, certificates are checked for several validation characteristics:

a) If the certificate 'notAfter' date is in the past, then this is an expired certificate which is considered invalid;

b) The certificate chain must terminate with a trusted CA certificate;

c) Server certificates consumed by the TOE TLS client must have a 'serverAuthentication' extendedKeyUsage purpose;

d) A trusted CA certificate is defined as any certificate loaded into the TOE trust store that has, at a minimum, a basicConstraints extension with the CA flag set to TRUE.

88 Certificate revocation checking is performed using OCSP on intermediate CA and leaf certificates at load time and hourly thereafter.

89 As X.509 certificates are not used for trusted updates, firmware integrity self-tests or client authentication, the code-signing and clientAuthentication purpose is not checked in the extendedKeyUsage for related certificates.

90 The X.509 certificates for each of the given scenarios are validated using the certificate path validation algorithm defined in RFC 5280, which can be summarized as follows:

a) The public key algorithm and parameters are checked

b) The current date/time is checked against the validity period revocation status is checked

c) Issuer name of X matches the subject name of X+1

d) Name constraints are checked

e) Policy OIDs are checked

f) Policy constraints are checked; issuers are ensured to have CA signing bits

g) Path length is checked

h) Critical extensions are processed

91 If, during the entire trust chain verification activity, any certificate under review fails a verification check, then the entire trust chain is deemed untrusted.

6.3.7 FIA_X509_EXT.2

92 The TOE has a single X509 trust store where all root CA and intermediate CA certificates can be stored and managed. The trust store is not cached: if a certificate

is deleted, it is immediately untrusted. If a certificate is added to the trust store, it is immediately trusted for its given scope.

93 Instructions for configuring the trusted IT entities to supply appropriate X.509 certificates are captured in the guidance documents.

94 As part of the verification process, OCSP is used to determine whether the certificate is revoked or not. If the OCSP responder cannot be contacted at certificate load time, then the TOE will reject the certificate. The OCSP service will periodically attempt to contact the OCSP responder according to a configurable interval (default is one hour).

6.3.8 FIA_X509_EXT.3

95 For the Certificate Signing Request, a CN is required and may be an IP address, or DNS name. An IP address is required in the SAN IP.1 field. Additional SANs may be specified by IP address, URI, DNS name or directory name.

6.4 Security Management

6.4.1 FMT_MOF.1/ManualUpdate

96 The TOE restricts the ability to perform software updates to a TestStream user with Administrator rights.

6.4.2 FMT_MOF.1/Functions

97 The TOE restricts the ability to modify (enable/disable) transmission of audit records to an external audit server to a TestStream user with Administrator rights and linux user ‘tsadmin’

6.4.3 FMT_MTD.1/CoreData

98 Users are required to login before being provided with access to any administrative functions.

99 The TOE restricts the ability to manage the TSF data to a TestStream user with Administrator rights and linux user ‘tsadmin’.

100 The TOE restricts the ability to manage the trust store and X.509v3 certificates to Administrators with the appropriate permissions.

6.4.4 FMT_MTD.1/CryptoKeys

101 The TOE restricts the ability to manage SSH, TLS and any configured X.509 private keys to linux user ‘tsadmin’.

6.4.5 FMT_SMF.1

102 The TOE may be managed via the CLI (console & SSH) or GUI (TLS). The specific management capabilities include:

a) Ability to administer the TOE locally and remotely (GUI, CLI)

b) Ability to configure the access banner (GUI, CLI)

c) Ability to configure the session inactivity time before session termination or locking (GUI, CLI)

d) Ability to update the TOE and to verify the updates (GUI)

e) Ability to configure the authentication failure parameters (GUI)

f) Ability to configure audit to external IT entity (enable/disable remote logging) (GUI, CLI)

g) Ability to manage the cryptographic keys (CLI)

h) Ability to re-enable administrator account (GUI, CLI)

i) Ability to configure the time which is used for time-stamps (GUI)

j) Ability to Manage the TOE's X.509v3 certificates and trust store (CLI)

k) Ability to import X.509v3 certificates to the TOE's trust store (CLI)

l) Ability to manage the trusted public keys database (CLI).

6.4.6 FMT_SMR.2

103 The TOE implements role-based access control and supports the following roles:

a) **Security Administrator**

104 Local and remote management of the TOE is restricted to Security Administrators.

105 For management and administration of the underlying Linux operating system, the security administrator must use the 'tsadmin' account.

6.5 Protection of the TSF

6.5.1 FPT_SKP_EXT.1

106 Keys are protected as described in Table 16. In all cases, plaintext keys cannot be viewed through an interface designed specifically for that purpose.

Table 16: Keys

Key/Password	Storage	Zeroization
TLS Private Keys	Persistent - plaintext	Overwritten with zeroes by administrator command.
TLS DH Keys	RAM - plaintext	Overwritten with zeroes upon termination of the session or reboot of the appliance
TLS Session Keys	RAM - plaintext	Overwritten with zeroes upon termination of the session or reboot of the appliance
TLS Session Authentication Keys	RAM - plaintext	Overwritten with zeroes upon termination of the session or reboot of the appliance
SSH Private Keys	Persistent - plaintext	Overwritten with zeroes by administrator command.
SSH DH Keys	RAM - plaintext	Overwritten with zeroes upon termination of the session or reboot of the appliance
SSH Session Keys	RAM - plaintext	Overwritten with zeroes upon termination of the session or reboot of the appliance

Key/Password	Storage	Zeroization
SSH Session Authentication Keys	RAM - plaintext	Overwritten with zeroes upon termination of the session or reboot of the appliance

6.5.2 FPT_APW_EXT.1

107 Passwords are protected as describe in Table 17. In all cases plaintext passwords cannot be viewed through an interface designed specifically for that purpose.

Table 17: Passwords

Key/Password	Generation/ Algorithm	Storage
Locally stored linux passwords	User generated	Persistent - SHA-512 hashed
TestStream passwords (postgresql DB)	User generated	Persistent - SHA-1 + AES128-CBC encrypted

6.5.3 FPT_TST_EXT.1

108 At startup, the TOE undergoes the following tests:

a) Integrity of firmware components are checked with a CRC-32 checksum against two copies located in NOR Flash, each with their own CRC-32 checksum. These components include the Linux kernel, and Linux init filesystem, which are checked before starting the Linux OS.

b) Known answer tests for implemented algorithms via the OpenSSL FIPS_test_suite.

c) Central Processing Unit (CPU) and Memory Basic Input/Output System (BIOS) self-tests – The U-Boot Bootloader initializes the CPU and the CPU parameters to access RAM. The RAM is then tested before the bootloader copies itself to it. The Firmware integrity is then checked as mentioned above and further initialization is performed if the firmware integrity test is successful.

109 These tests ensure the correct operation of the cryptographic functionality of the TOE, the CPU and BIOS and verify firmware integrity. If the CPU, or BIOS tests fail, the device will not complete the boot up operation. Any failure in the firmware integrity tests will prevent the TOE from starting up.

110 The cryptographic module executes the following conditional tests when the related service is invoked:

a) Continuous DRBG health tests implemented by Jitter RNG which include the following:

i) Stuck test

ii) Repetition count test

iii) Adaptive proportion test

6.5.4 FPT_TUD_EXT.1

111 The current TOE software version may be queried using the CLI or GUI.

112 The administrator downloads software updates from the my.netscout.com website along with a SHA256 file containing the published hash.

113 The Security Administrator must manually verify the that the published hash matches a SHA256 hash of the update file before initiating an update.

6.5.5 FPT_STM_EXT.1

114 The TOE incorporates an internal clock that is used to maintain date and time. The Security Administrator sets the date and time during initial TOE configuration and may change the time during operation.

115 The TOE makes used of time for the following:

a) Audit record timestamps

b) Session timeouts (lockout enforcement)

c) Certificate validation

6.6 TOE Access

6.6.1 FTA_SSL_EXT.1

116 The Security Administrator may configure the TOE to terminate an inactive local interactive session (CLI) following a specified period of time. The timeout value is disabled by default.

6.6.2 FTA_SSL.3

117 The Security Administrator may configure the TOE to terminate an inactive remote interactive session via CLI (console, SSH) and TestStream Management GUI following a specified period of time.

118 The default keep-alive timeout interval for the REST API is 15 minutes.

6.6.3 FTA_SSL.4

119 Administrative users may terminate their own sessions at any time.

6.6.4 FTA_TAB.1

120 Once enabled, the TOE displays an administrator configurable message to all users prior to login at interactive interfaces including the CLI (console, SSH), TestStream Management GUI, and the REST API.

6.7 Trusted Path/Channels

6.7.1 FTP_ITC.1

121 The TOE supports SSHv2 with ECDSA public keys to protect communications between itself and the following IT entities:

a) Syslog audit server.

122 The TOE also supports AES 128 and 256-bit Counter mode encryption in addition to the characteristics described in FCS_SSHC_EXT.1.

6.7.2 FTP_TRP.1/Admin

123 The TOE provides the following trusted paths for remote administration:

a) **Remote CLI (SSHv2)**. Remotely connecting via SSH protocol on port 22 (operating system functions, initial TOE configuration, and maintenance operations), and port 22022 (TestStream CLI for administrator use).

b) **TestStream Management GUI (TLS).** Remotely connecting to the TOE via TLS using the TestStream Management GUI Java applet.

c) **REST API (HTTPS).** Programmatic administration of TestStream functionality.

7 Rationale

7.1 Conformance Claim Rationale

124 The following rationale is presented with regard to the PP conformance claims:

a) **TOE type.** As identified in section 2.1, the TOE is network device, consistent with the NDcPP.

b) **Security problem definition.** As shown in section 3, the threats, OSPs and assumptions are reproduced directly from the NDcPP.

c) **Security objectives.** As shown in section 4, the security objectives are reproduced directly from the NDcPP.

d) **Security requirements.** As shown in section 5, the security requirements are reproduced directly from the NDcPP. No additional requirements have been specified.

7.2 Security Objectives Rationale

125 All security objectives are drawn directly from the NDcPP.

7.3 Security Requirements Rationale

126 All security requirements are drawn directly from the NDcPP. Table 18 presents a mapping between threats and SFRs as presented in the NDcPP.

Table 18: NDcPP SFR Rationale

Identifier	SFR Rationale
T.UNAUTHORIZED_ ADMINISTRATOR_ACCESS	• The Administrator role is defined in FMT_SMR.2 and the relevant administration capabilities are defined in FMT_SMF.1 and FMT_MTD.1/CoreData, with optional additional capabilities in FMT_MOF.1/Services and FMT_MOF.1/Functions • The actions allowed before authentication of an Administrator are constrained by FIA_UIA_EXT.1, and include the advisory notice and consent warning message displayed according to FTA_TAB.1 • The requirement for the Administrator authentication process is described in FIA_UAU_EXT.2 • Locking of Administrator sessions is ensured by FTA_SSL_EXT.1 (for local sessions), FTA_SSL.3 (for remote sessions), and FTA_SSL.4 (for all interactive sessions) • The secure channel used for remote Administrator connections is specified in FTP_TRP.1/Admin • (Malicious actions carried out from an Administrator session are separately addressed by T.UNDETECTED_ACTIVITY)

Identifier	SFR Rationale
	• (Protection of the Administrator credentials is separately addressed by T.PASSWORD_CRACKING).
T.WEAK_CRYPTOGRAPHY	• Requirements for key generation and key distribution are set in FCS_CKM.1 and FCS_CKM.2 respectively • Requirements for use of cryptographic schemes are set in FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, and FCS_COP.1/KeyedHash • Requirements for random bit generation to support key generation and secure protocols (see SFRs resulting from T.UNTRUSTED_COMMUNICATION_CHANNELS) are set in FCS_RBG_EXT.1 • Management of cryptographic functions is specified in FMT_SMF.1
T.UNTRUSTED_ COMMUNICATION_ CHANNELS	• The general use of secure protocols for identified communication channels is described at the top level in FTP_ITC.1 and FTP_TRP.1/Admin; for distributed TOEs the requirements for inter-component communications are addressed by the requirements in FPT_ITT.1 • Requirements for the use of secure communication protocols are set for all the allowed protocols in FCS_DTLSC_EXT.1, FCS_DTLSC_EXT.2, FCS_DTLSS_EXT.1, FCS_DTLSS_EXT.2, FCS_HTTPS_EXT.1, FCS_IPSEC_EXT.1, FCS_SSHC_EXT.1, FCS_SSHS_EXT.1, FCS_TLSC_EXT.1, FCS_TLSC_EXT.2, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2 • Optional and selection-based requirements for use of public key certificates to support secure protocols are defined in FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3
T.WEAK_ AUTHENTICATION_ ENDPOINTS	• The use of appropriate secure protocols to provide authentication of endpoints (as in the SFRs addressing T.UNTRUSTED_COMMUNICATION_CHANNELS) are ensured by the requirements in FTP_ITC.1 and FTP_TRP.1/Admin; for distributed TOEs the authentication requirements for endpoints in inter-component communications are addressed by the requirements in FPT_ITT.1 • Additional possible special cases of secure authentication during registration of distributed TOE components are addressed by FCO_CPC_EXT.1 and FTP_TRP.1/Join.
T.UPDATE_COMPROMISE	• Requirements for protection of updates are set in FPT_TUD_EXT.1 • Additional optional use of certificate-based protection of signatures can be specified using FPT_TUD_EXT.2, supported by the X.509 certificate processing requirements in FIA_X509_EXT.1, FIA_X509_EXT.2 and FIA_X509_EXT.3

Identifier	SFR Rationale
	• Requirements for management of updates are defined in FMT_SMF.1 and (for manual updates) in FMT_MOF.1/ManualUpdate,
T.UNDETECTED_ACTIVITY	• Requirements for basic auditing capabilities are specified in FAU_GEN.1 and FAU_GEN.2, with timestamps provided according to FPT_STM_EXT.1 • Requirements for protecting audit records stored on the TOE are specified in FAU_STG.1 • Requirements for secure transmission of local audit records to an external IT entity via a secure channel are specified in FAU_STG_EXT.1 • If (optionally) configuration of the audit functionality is provided by the TOE then this is specified in FMT_SMF.1, and confining this functionality to Security Administrators is required by FMT_MOF.1/Functions.
T.SECURITY_ FUNCTIONALITY_ COMPROMISE	• Protection of secret/private keys against compromise is specified in FPT_SKP_EXT.1 • Secure destruction of keys is specified in FCS_CKM.4 • If (optionally) management of keys is provided by the TOE then this is specified in FMT_SMF.1, and confining this functionality to Security Administrators is required by FMT_MTD.1/CryptoKeys • (Protection of passwords is separately covered under T.PASSWORD_CRACKING)
T.PASSWORD_CRACKING	• Requirements for password lengths and available characters are set in FIA_PMG_EXT.1 • Protection of password entry by providing only obscured feedback is specified in FIA_UAU.7 • Actions on reaching a threshold number of consecutive password failures are specified in FIA_AFL.1 • Requirements for secure storage of passwords are set in FPT_APW_EXT.1.
T.SECURITY_ FUNCTIONALITY_FAILURE	• Requirements for running self-test(s) are defined in FPT_TST_EXT.1 • Optional use of certificates to support self-test(s) is defined in FPT_TST_EXT.2 (with support for the use of certificates in FIA_X509_EXT.1, FIA_X509_EXT.2, and FIA_X509_EXT.3),