Infinera Corporation Transcend Network Management System Client 18.10.3 Security Target

Version 1.5
12/06/2022

Prepared for:

Infinera Corporation

9005 Junction Dr, Suite C
Annapolis Junction, MD 20701

Prepared By:

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 Infinera Corporation Transcend Network Management System Client provided by Infinera Corporation. The TOE is being evaluated as a software application.

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 – Infinera Corporation Transcend Network Management System Client 18.10.3 Security Target

ST Version – Version 1.5

ST Date – 12/06/2022

1.2 TOE Reference

TOE Identification – Infinera Corporation Transcend Network Management System Client 18.10.3

TOE Developer – Infinera Corporation

Evaluation Sponsor – Infinera Corporation

1.3 TOE Overview

The Target of Evaluation (TOE) is Transcend Network Management System (TNMS) Client version 18.10.3.

Note that the full TNMS system consists of client and server components, this evaluation is limited only to the client components since the Protection Profile for Application Software, Version 1.4, 07 October 2021 (ASPP14) does not support the evaluation of distributed application solutions. Please refer to the *Infinera Corporation Transcend Network Management System Server (ASPP14/PKGTLS11) Security Target* where the other TNMS components are addressed.

1.4 TOE Description

The Transcend Network Management System (TNMS) is designed to provide end-to-end network and service management across multiple technologies and equipment vendors. The TNMS system has been developed with the following features:

- Network and service management with a broad feature set, from deep node-level troubleshooting to end-to-end service configuration and monitoring, across multiple technologies and equipment vendors for improved operational efficiency.
- An advanced graphical user interface providing overviews of the most relevant data and guidance through configuration steps.
- Flexible deployment configuration options that fit various needs of different network operators: from small networks to very large infrastructures, including support for virtualization and high availability.
- Clear user interfaces that provide user-friendly navigation and supporting responsive and efficient use.

For purposes of this evaluation, the TNMS Client is a software application that offers a secure web-based user interface to its users and in turn securely communicates management instructions to a configured TNMS Server. This evaluation addresses and is limited to the security functions claimed in Section 5 and further described in Section 6 of this Security Target (ST).

1.4.1 TOE Architecture

Software Requirements:

The TNMS Client is a single Java application designed to run in the following operational environment:

- Microsoft Windows 10 (64 bit) on a 64 bit Intel Xeon processor
- Amazon Corretto (OpenJDK) JDK/JRE 11.0.6
- Oracle Java JRE 8u201

The TNMS Client provides a graphical user-interface to its users so that they can perform management functions. The TNMS Client implements a TLS client to connect to the TNMS Server so that management instructions can be processed and forwarded as necessary to network entities.

The secure communication features all utilize Bouncy Castle (installed as part of the TNSM Client) for cryptographic operations.

The TOE was installed and tested on the following platform

Microsoft Windows 10 Home (64 bit) on Intel® Xeon® E5-2670 with Amazon Corretto JDK 11.0.6 and Oracle Java JRE 8u201

1.4.1.1 Physical Boundaries

The TNMS Client TOE is a single Java application running on Amazon Corretto (OpenJDK) JDK/JRE version 11.0.6 and Oracle Java JRE 8u201 JRE operating in a Microsoft Windows 10 (64 bit) environment. The only external services the TOE depends on are a Certificate Authority for X509 certificate validation services and the TNMS Server with which the TNMS Client is configured to work.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by TNMS:

- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Privacy
- Protection of the TSF
- Trusted path/channels

1.4.1.2.1 Cryptographic support

The TOE uses Automated Cryptographic Validation Test System (ACVTS)-validated cryptographic algorithm implementations, provided by its Bouncy Castle cryptographic module installed with the TOE, to support asymmetric key generation, encryption/decryption, signature generation and verification and establishment of trusted channels to protect data in transit. The TOE implements a TLS client to securely communicate with a TNMS Server. The TOE also relies on the underlying Java Runtime Environment to generate entropy that is used as input data for the TOE's deterministic random bit generator (DRBG).

1.4.1.2.2 User data protection

The TOE does not access any hardware resources or sensitive information repositories and no sensitive data is stored in non-volatile memory. Inbound and outbound network communications are restricted to those that are user-initiated.

1.4.1.2.3 Identification and authentication

The TOE implements X509 certificate validation to validate the revocation status of certificates using CRL. The TOE uses X509 certificates to support TLS authentication.

1.4.1.2.4 Security management

The TOE provides a graphical user interface to operate the TOE.

The TOE requires no management beyond configuration of the URI of the TNMS server to which it connects and the root CA certificate to which the TNMS server's peer certificate should chain.

When configured with default credentials or no credentials, the TOE restricts its functionality and only allows the ability to set new credentials. By default, the TOE is configured with file permissions to protect itself and its data from unauthorized access.

1.4.1.2.5 Privacy

The TOE does not transmit personally identifiable information (PII) over any network interfaces.

1.4.1.2.6 Protection of the TSF

The TOE protects itself against exploitation by implementing address space layout randomization (ASLR) and by not allocating any memory region for both write and execute permission. The TOE uses standard platform APIs and includes a number of third party libraries used to perform its functions.

The TOE includes mechanisms to check for updates and to query the current version of the application software. TOE software is digitally signed and distributed using the platform-supported package manager. The TOE does not update its own binary code in any way and when removed, all traces of the TOE application software are deleted.

1.4.1.2.7 Trusted path/channels

The TOE protects communications between itself and remote administrators using TLS and between itself and the TNMS Server using TLS.

1.4.2 TOE Documentation

The following user and administrative guidance is available:

Infinera Transcend Network Management System Client 18.10.3 Administrative Guidance for Common Criteria, Version 1.2, December 6, 2022

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 Extended
- Package Claims:
  - Protection Profile for Application Software, Version 1.4, 10/7/2021
  - Functional Package for Transport Layer Security (TLS), Version 1.1, 2/12/2019 (ASPP14/PKGTLS11)

Package	Technical Decision	Applied	Notes
PP_APP_V1.4	TD0669 – FIA_X509_EXT.1 Test 4 Interpretation	Yes	Applied
PP_APP_V1.4	TD0664 – Testing activity for FPT_TUD_EXT.2.2	Yes	Applied
PP_APP_V1.4	TD0659 – Change to Required NIST Curves for FCS_CKM.1/AK	Yes	Applied
PP_APP_V1.4	TD0655 – Mutual authentication in FTP_DIT_EXT.1 for SW App	Yes	Applied
PP_APP_V1.4	TD0650- Conformance claim sections updated to allow for MOD_VPNC_V2.3 and 2.4	No	VPNC not claimed
PP_APP_V1.4	TD0628 – Addition of Container Image to Package Format	Yes	Applied
PP_APP_V1.4	TD0626 – FCS_COP.1 Keyed Hash Selections	Yes	Applied
PP_APP_V1.4	TD0624 – Addition of DataStore for Stored and Setting Configuration Options	Yes	Applied
PKG_TLS_V1.1	TD0588 - Session Resumption Support in TLS package	No	Not supported
PKG_TLS_V1.1	TD0513 - CA Certificate loading	Yes	Applied
PKG_TLS_V1.1	TD0499 - Testing with pinned certificates	No	Not supported
PKG_TLS_V1.1	TD0469 - Modification of test activity for FCS_TLSS_EXT.1.1 test 4.1	No	Not Supported
PKG_TLS_V1.1	TD0442 - Updated TLS Ciphersuites for TLS Package	Yes	Applied

2.1 Conformance Rationale

The ST conforms to the ASPP14/PKGTLS11. 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 ASPP14/PKGTLS11 and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The ASPP14/PKGTLS11 offers additional information about the identified security objectives, but that has not been reproduced here and the ASPP14/PKGTLS11 should be consulted if there is interest in that material.

In general, the ASPP14/PKGTLS11 has defined Security Objectives appropriate for software application and as such are applicable to the Infinera Corporation Transcend Network Management System Client TOE.

3.1 Security Objectives for the Operational Environment

OE.PLATFORM The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and any discrete execution environment provided to the TOE.

OE.PROPER_ADMIN The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.

OE.PROPER_USER The user of the application software is not willfully negligent or hostile, and uses the software within compliance of the applied enterprise security policy.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the ASPP14/PKGTLS11. The ASPP14/PKGTLS11 defines the following extended requirements and since they are not redefined in this ST the ASPP14/PKGTLS11 should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- ASPP14:FCS_RBG_EXT.1: Random Bit Generation Services
- ASPP14:FCS_RBG_EXT.2: Random Bit Generation from Application
- ASPP14:FCS_STO_EXT.1: Storage of Credentials
- PKGTLS11:FCS_TLS_EXT.1: TLS Protocol
- PKGTLS11:FCS_TLSC_EXT.1: TLS Client Protocol
- PKGTLS11:FCS_TLSC_EXT.5: TLS Client Support for Supported Groups Extension
- ASPP14:FDP_DAR_EXT.1: Encryption Of Sensitive Application Data
- ASPP14:FDP_DEC_EXT.1: Access to Platform Resources
- ASPP14:FDP_NET_EXT.1: Network Communications
- ASPP14:FIA_X509_EXT.1: X.509 Certificate Validation
- ASPP14:FIA_X509_EXT.2: X.509 Certificate Authentication
- ASPP14:FMT_CFG_EXT.1: Secure by Default Configuration
- ASPP14:FMT_MEC_EXT.1: Supported Configuration Mechanism
- ASPP14:FPR_ANO_EXT.1: User Consent for Transmission of Personally Identifiable
- ASPP14:FPT_AEX_EXT.1: Anti-Exploitation Capabilities
- ASPP14:FPT_API_EXT.1: Use of Supported Services and APIs
- ASPP14:FPT_IDV_EXT.1: Software Identification and Versions
- ASPP14:FPT_LIB_EXT.1: Use of Third Party Libraries
- ASPP14:FPT_TUD_EXT.1: Integrity for Installation and Update
- ASPP14:FPT_TUD_EXT.2: Integrity for Installation and Update
- ASPP14:FTP_DIT_EXT.1: Protection of Data in Transit

Extended SARs:

- ALC_TSU_EXT.1: Timely Security Updates

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 ASPP14/PKGTLS11. The refinements and operations already performed in the ASPP14/PKGTLS11 are not identified (e.g., highlighted) here, rather the requirements have been copied from the ASPP14/PKGTLS11 and any residual operations have been completed herein. Of particular note, the ASPP14/PKGTLS11 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 ASPP14/PKGTLS11. The ASPP14/PKGTLS11 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by Infinera Corporation Transcend Network Management System Client TOE.

Requirement Class	Requirement Component
FCS: Cryptographic support	ASPP14:FCS_CKM.1: Cryptographic Key Generation Services
	ASPP14:FCS_CKM.1/AK: Cryptographic Asymmetric Key Generation
	ASPP14:FCS_CKM.1/SK: Cryptographic Symmetric Key Generation
	ASPP14:FCS_CKM.2: Cryptographic Key Establishment
	ASPP14:FCS_COP.1/Hash: Cryptographic Operation - Hashing
	ASPP14:FCS_COP.1/KeyedHash: Cryptographic Operation - Keyed-Hash Message Authentication
	ASPP14:FCS_COP.1/Sig: Cryptographic Operation - Signing
	ASPP14:FCS_COP.1/SKC: Cryptographic Operation - Encryption/Decryption
	ASPP14:FCS_RBG_EXT.1: Random Bit Generation Services
	ASPP14:FCS_RBG_EXT.2: Random Bit Generation from Application
	ASPP14:FCS_STO_EXT.1: Storage of Credentials
	PKGTLS11:FCS_TLS_EXT.1: TLS Protocol
	PKGTLS11:FCS_TLSC_EXT.1: TLS Client Protocol
	PKGTLS11:FCS_TLSC_EXT.5: TLS Client Support for Supported Groups Extension
FDP: User data protection	ASPP14:FDP_DAR_EXT.1: Encryption Of Sensitive Application Data
	ASPP14:FDP_DEC_EXT.1: Access to Platform Resources
	ASPP14:FDP_NET_EXT.1: Network Communications
FIA: Identification and authentication	ASPP14:FIA_X509_EXT.1: X.509 Certificate Validation
	ASPP14:FIA_X509_EXT.2: X.509 Certificate Authentication
FMT: Security management	ASPP14:FMT_CFG_EXT.1: Secure by Default Configuration
	ASPP14:FMT_MEC_EXT.1: Supported Configuration Mechanism
	ASPP14:FMT_SMF.1: Specification of Management Functions
FPR: Privacy	ASPP14:FPR_ANO_EXT.1: User Consent for Transmission of Personally Identifiable

FPT: Protection of the TSF	ASPP14:FPT_AEX_EXT.1: Anti-Exploitation Capabilities
	ASPP14:FPT_API_EXT.1: Use of Supported Services and APIs
	ASPP14:FPT_IDV_EXT.1: Software Identification and Versions
	ASPP14:FPT_LIB_EXT.1: Use of Third Party Libraries
	ASPP14:FPT_TUD_EXT.1: Integrity for Installation and Update
	ASPP14:FPT_TUD_EXT.2: Integrity for Installation and Update
FTP: Trusted path/channels	ASPP14:FTP_DIT_EXT.1: Protection of Data in Transit

Table 1 TOE Security Functional Components

5.1.1 Cryptographic support (FCS)

5.1.1.1 Cryptographic Key Generation Services (ASPP14:FCS_CKM.1)

ASPP14:FCS_CKM.1.1

The application shall [***implement asymmetric key generation***].

5.1.1.2 Cryptographic Asymmetric Key Generation (ASPP14:FCS_CKM.1/AK)

ASPP14:FCS_CKM.1.1/AK

The application shall [***implement functionality***] to generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm [***/RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.3, ECC schemes using 'NIST curves' P-384 and [P-256] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4***].

5.1.1.3 Cryptographic Symmetric Key Generation (ASPP14:FCS_CKM.1/SK)

ASPP14:FCS_CKM.1.1/SK

The application shall generate symmetric cryptographic keys using a Random Bit Generator as specified in FCS_RBG_EXT.1 and specified cryptographic key sizes [***128 bit, 256 bit***].

5.1.1.4 Cryptographic Key Establishment (ASPP14:FCS_CKM.2)

ASPP14:FCS_CKM.2.1

The application shall [***implement functionality***] to perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [***RSA-based key establishment schemes that meets the following: NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography", Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography'***].

5.1.1.5 Cryptographic Operation - Hashing (ASPP14:FCS_COP.1/Hash)

ASPP14:FCS_COP.1.1/Hash

The application shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-1, SHA-256, SHA-384, SHA-512***] and message digest sizes [***160, 256, 384, 512***] bits that meet the following: FIPS Pub 180-4.

5.1.1.6 Cryptographic Operation - Keyed-Hash Message Authentication (ASPP14:FCS_COP.1/KeyedHash)

ASPP14:FCS_COP.1.1/KeyedHash

The application shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***] and [***HMAC-SHA-1***] with key sizes [**160, 256, 384, 512**] and message digest sizes [***256, 384, 512***] and [***160***] bits that meet the following: FIPS Pub 198-1, 'The Keyed-Hash Message Authentication Code' and FIPS Pub 180-4, 'Secure Hash Standard'. (TD0626 applied)

5.1.1.7 Cryptographic Operation - Signing (ASPP14:FCS_COP.1/Sig)

ASPP14:FCS_COP.1.1/Sig

The application shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [
RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 4,
ECDSA schemes using 'NIST curves' P-256, P-384 and [P-521] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 5].

5.1.1.8 Cryptographic Operation - Encryption/Decryption (ASPP14:FCS_COP.1/SKC)

ASPP14:FCS_COP.1.1/SKC

The application shall perform encryption/decryption in accordance with a specified cryptographic algorithm
[***AES-CBC (as defined in NIST SP 800-38A) mode, AES-GCM (as defined in NIST SP 800-38D) mode***]
and cryptographic key sizes [***128-bit, 256-bit***].

5.1.1.9 Random Bit Generation Services (ASPP14:FCS_RBG_EXT.1)

ASPP14:FCS_RBG_EXT.1.1

The application shall [***invoke platform-provided DRBG functionality, implement DRBG functionality***] for its cryptographic operations.

5.1.1.10 Random Bit Generation from Application (ASPP14:FCS_RBG_EXT.2)

ASPP14:FCS_RBG_EXT.2.1

The application shall perform all deterministic random bit generation (DRBG) services in accordance with NIST Special Publication 800-90A using [***Hash_DRBG (any)***].

ASPP14:FCS_RBG_EXT.2.2

The deterministic RBG shall be seeded by an entropy source that accumulates entropy from a platform-based DRBG and [***no other noise source***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength (according to NIST SP 800-57) of the keys and hashes that it will generate.

5.1.1.11 Storage of Credentials (ASPP14:FCS_STO_EXT.1)

ASPP14:FCS_STO_EXT.1.1

The application shall [***not store any credentials***] to non-volatile memory.

5.1.1.12 TLS Protocol (PKGTLS11:FCS_TLS_EXT.1)

PKGTLS11:FCS_TLS_EXT.1.1

The product shall implement [***TLS as a client,***]

5.1.1.13 TLS Client Protocol (PKGTLS11:FCS_TLSC_EXT.1)

PKGTLS11:FCS_TLSC_EXT.1.1
The product shall implement TLS 1.2 (RFC 5246) and [***no earlier TLS versions***] as a client that supports the cipher suites [***TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246, TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246, TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246, TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246, TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288, TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289, 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, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289***] and also supports functionality for [***none***] (TD0442 applied)

PKGTLS11:FCS_TLSC_EXT.1.2
The product shall verify that the presented identifier matches the reference identifier according to RFC 6125.

PKGTLS11:FCS_TLSC_EXT.1.3
The product shall not establish a trusted channel if the server certificate is invalid [***with no exceptions***]

5.1.1.14 TLS Client Support for Supported Groups Extension (PKGTLS11:FCS_TLSC_EXT.5)

PKGTLS11:FCS_TLSC_EXT.5.1
The product shall present the Supported Groups Extension in the Client Hello with the supported groups [***secp256r1, secp384r1, secp521r1,***]

5.1.2 User data protection (FDP)

5.1.2.1 Encryption Of Sensitive Application Data (ASPP14:FDP_DAR_EXT.1)

ASPP14:FDP_DAR_EXT.1.1
The application shall [***not store any sensitive data***] in non-volatile memory.

5.1.2.2 Access to Platform Resources (ASPP14:FDP_DEC_EXT.1)

ASPP14:FDP_DEC_EXT.1.1
The application shall restrict its access to [***network connectivity***].

ASPP14:FDP_DEC_EXT.1.2
The application shall restrict its access to [***no sensitive information repositories***].

5.1.2.3 Network Communications (ASPP14:FDP_NET_EXT.1)

ASPP14:FDP_NET_EXT.1.1
The application shall restrict network communication to [***user-initiated communication for [connecting to the TNMS Server to facilitate management functions]***].

5.1.3 Identification and authentication (FIA)

5.1.3.1 X.509 Certificate Validation (ASPP14:FIA_X509_EXT.1)

ASPP14:FIA_X509_EXT.1.1

The application shall [***implement functionality***] to validate certificates in accordance with the following rules:
- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The application shall validate a certificate path by ensuring the presence of the basicConstraints extension, that the CA flag is set to TRUE for all CA certificates, and that any path constraints are met.
- The application shall validate that any CA certificate includes caSigning purpose in the key usage field
- The application shall validate the revocation status of the certificate using [***CRL as specified in RFC 5280 Section 6.3, CRL as specified in RFC 8603***]
- The application shall validate the extendedKeyUsage (EKU) field according to the following rules:
  - o 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.
  - o 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 EKU field.
  - o 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 EKU field.
  - o S/MIME certificates presented for email encryption and signature shall have the Email Protection purpose (id-kp 4 with OID 1.3.6.1.5.5.7.3.4) in the EKU field.
  - o 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 EKU field.
  - o Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kpcmcRA with OID 1.3.6.1.5.5.7.3.28) in the EKU field.

ASPP14:FIA_X509_EXT.1.2

The application shall treat a certificate as a CA certificate only if the basicConstraints extension is present and the CA flag is set to TRUE.

5.1.3.2 X.509 Certificate Authentication (ASPP14:FIA_X509_EXT.2)

ASPP14:FIA_X509_EXT.2.1

The application shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [***TLS***].

ASPP14:FIA_X509_EXT.2.2

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

5.1.4 Security management (FMT)

5.1.4.1 Secure by Default Configuration (ASPP14:FMT_CFG_EXT.1)

ASPP14:FMT_CFG_EXT.1.1

The application shall provide only enough functionality to set new credentials when configured with default credentials or no credentials.

ASPP14:FMT_CFG_EXT.1.2

The application shall be configured by default with file permissions which protect the application's binaries and data files from modification by normal unprivileged user.

5.1.4.2 Supported Configuration Mechanism (ASPP14:FMT_MEC_EXT.1)

ASPP14:FMT_MEC_EXT.1.1
The application shall [***invoke the mechanisms recommended by the platform vendor for storing and setting configuration options***].

5.1.4.3 Specification of Management Functions (ASPP14:FMT_SMF.1)

ASPP14:FMT_SMF.1.1
The TSF shall be capable of performing the following management functions [***[specify the TNMS server (by FQDN or IP address), manage trusted root CA certificates, connect with to the TNMS server]***].

5.1.5 Privacy (FPR)

5.1.5.1 User Consent for Transmission of Personally Identifiable (ASPP14:FPR_ANO_EXT.1)

ASPP14:FPR_ANO_EXT.1.1
The application shall [***not transmit PII over a network***].

5.1.6 Protection of the TSF (FPT)

5.1.6.1 Anti-Exploitation Capabilities (ASPP14:FPT_AEX_EXT.1)

ASPP14:FPT_AEX_EXT.1.1
The application shall not request to map memory at an explicit address except for [**no exceptions**].
ASPP14:FPT_AEX_EXT.1.2
The application shall [***not allocate any memory region with both write and execute permissions***].
ASPP14:FPT_AEX_EXT.1.3
The application shall be compatible with security features provided by the platform vendor.
ASPP14:FPT_AEX_EXT.1.4
The application shall not write user-modifiable files to directories that contain executable files unless explicitly directed by the user to do so.
ASPP14:FPT_AEX_EXT.1.5
The application shall be built with stack-based buffer overflow protection enabled.

5.1.6.2 Use of Supported Services and APIs (ASPP14:FPT_API_EXT.1)

ASPP14:FPT_API_EXT.1.1
The application shall use only documented platform APIs.

5.1.6.3 Software Identification and Versions (ASPP14:FPT_IDV_EXT.1)

ASPP14:FPT_IDV_EXT.1.1
The application shall be versioned with [***[major and minor version and build number]***]

5.1.6.4 Use of Third Party Libraries (ASPP14:FPT_LIB_EXT.1)

ASPP14:FPT_LIB_EXT.1.1
The application shall be packaged with only [***the following 3rd party components***]

Component name	Component version
Apache Xerces	J2.12.0.SP02

	J2.12.0.SP03
AdventNet	1.2.2
AOP Alliance (Java/J2EE AOP standard)	1.0
Apache Active MQ	5.17.2
Apache Avalon	4.3.1
Apache Commons IO	2.6 2.7
Apache Commons Lang	2.6 3.8
Apache Commons Logging	1.0.3 1.2
Apache Commons Math	3.6.1
Apache Commons Net	2.0 2.2 3.6
Apache Commons Text	1.4
Apache Commons Validator	1.6
Apache FOP	2.3 2.6
Apache FTP server	1.0.5
Apache HttpClient	3.1 4.56
Apache ORO	2.0.8
Apache Tomcat	9.0.54 9.0.65
Apache Velocity	2.0
Apache XML Commons	1.4.01
Apache XML Graphics Commons	2.3 2.6
Apached Commons Codec	1.10 1.11
ASM	6.2.1
AspectJ	1.9.5
Batik XML Util Library	1.10 1.14 1.16
Bouncy Castle Provider - FIPS	1.0.2 1.0.2.1 1.0.3 1.0.4 1.0.5
Bouncy Castle TLS/JSSE APIs (FIPS Distribution)	1.0.12.2 1.0.9

Castor	1.4.1
Cglib	3.2.8
Disruptor Framework	3.4.2
Docking Frames	1.1.12
Dom4j	1.6.1 2.1.1 2.1.3
EdDSA	0.3.0
edftpj	2.0.3
EHCache	2.10.5
Ganymed SSH for Java	2.2
Google Guice	4.0.23
google-collection	1.0
google-gson	2.1 2.8.9
Guava Google Core Libraries for Java	26
image4j	0.7
InstallAnywhere	2018 SP1
io.grpc	1.25
istack common utiltiy code runtime	3.0.8
Jakarta Activation API	1.2.1
Jakarta Mail	1.4
Java Architecture for XML Binding	2.3.2
Java Communications	1.0
Java FX	11.0.2
Java Swing	1.6.4
javaee/glassfish V2 Milestone	Build 19
JavaHelp	2.0
JavaHelp Search	2.0
Javax Inject from the JSR-330 Expert Group	1.0
JavaZoom Basic Player	3.0
JaxB Runtime	2.3.2
JCalendar	1.3.3
JDOM	1.1.3
Jersey	2.22.1
JFreeChart	1.0.13 1.5.0
JFreeReport	0.8.7 0.8.8-04

jgraphx	1.5.1.3
JIDE	2.2.4
Jlayer	1.0.1-1
Jscape	8.5.0
JSch	0.1.55
JSR305	3.0.2
JUnit	1.4.1 3.8.1
jzlib	1.1.3
Log4J	1.2.14 1.2.17
MiGLayout	5.2
MP3SPI	1.9.5-1
OpenCensus	0.25.0
OpenMap	5.1.15
OpenProps	0.8.5
Oracle Database JDBC Drivers	12.2.0.1
Plexus	2.7.1
reactive-streams	1.0.2
reflections	0.9.11
rxjava	2.2.2
SLF4J	1.7.13 1.7.25 1.7.26 1.7.30 1.7.31
snappy	0.3
SNMP4j	2.5.0 2.5.6
SSHJ	0.27.0
Tritonus	0.3.7-2
VorbisSPI	1.0.3-1
VT Dictionary Libraries	3.0
VT Password Libraries	3.1.2
webdavilb	2.0
Wildfly	15.0
Xalan	2.7.2
XBean :: Reflect	3.7
XML Commons External Components XML APIs Extensions	1.3.04

].

5.1.6.5 Integrity for Installation and Update (ASPP14:FPT_TUD_EXT.1)

ASPP14:FPT_TUD_EXT.1.1
The application shall [***provide the ability***] to check for updates and patches to the application software.
ASPP14:FPT_TUD_EXT.1.2
The application shall [***provide the ability***] to query the current version of the application software.
ASPP14:FPT_TUD_EXT.1.3
The application shall not download, modify, replace or update its own binary code.
ASPP14:FPT_TUD_EXT.1.4
Application updates shall be digitally signed such that the application platform can cryptographically verify them prior to installation.
ASPP14:FPT_TUD_EXT.1.5
The application is distributed [***as an additional software package to the platform OS***] .

5.1.6.6 Integrity for Installation and Update (ASPP14:FPT_TUD_EXT.2)

ASPP14:FPT_TUD_EXT.2.1
The application shall be distributed using the format of the platform-supported package manager.
ASPP14:FPT_TUD_EXT.2.2
The application shall be packaged such that its removal results in the deletion of all traces of the application, with the exception of configuration settings, output files, and audit/log events
ASPP14:FPT_TUD_EXT.2.3
The application installation package shall be digitally signed such that its platform can cryptographically verify them prior to installation.

5.1.7 Trusted path/channels (FTP)

5.1.7.1 Protection of Data in Transit (ASPP14:FTP_DIT_EXT.1)

ASPP14:FTP_DIT_EXT.1.1
The application shall [***encrypt all transmitted [data] with [TLS as a client as defined in the Functional Package for TLS]***] between itself and another trusted IT product.

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
	ALC_TSU_EXT.1: Timely Security Updates
ATE: Tests	ATE_IND.1: Independent Testing - Conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability Survey

Table 2 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 application shall be labelled with a 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 description in the TSS of how timely security updates are made to the TOE. Application developers must support updates to their products for purposes of fixing security vulnerabilities.

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.

5.2.3.3 Timely Security Updates (ALC_TSU_EXT.1)

ALC_TSU_EXT.1.1d
The developer shall provide a description in the TSS of how timely security updates are made to the TOE. Note: Application developers must support updates to their products for purposes of fixing security vulnerabilities.

ALC_TSU_EXT.1.2d
The developer shall provide a description in the TSS of how users are notified when updates change security properties or the configuration of the product.

ALC_TSU_EXT.1.1c
The description shall include the process for creating and deploying security updates for the TOE software.

ALC_TSU_EXT.1.2c
The description shall express the time window as the length of time, in days, between public disclosure of a vulnerability and the public availability of security updates to the TOE.

ALC_TSU_EXT.1.3c
The description shall include the mechanisms publicly available for reporting security issues pertaining to the TOE.

Note: The reporting mechanism could include web sites, email addresses, as well as a means to protect the sensitive nature of the report (e.g., public keys that could be used to encrypt the details of a proof-of-concept exploit).

ALC_TSU_EXT.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:

- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Privacy
- Protection of the TSF
- Trusted path/channels

6.1 Cryptographic support

ASPP14:FCS_CKM.1:

The TOE generates asymmetric RSA and ECDH keys during TLS connections to the TNMS server

ASPP14:FCS_CKM.1/AK:

The TOE generates 2048-bit RSA keys and P-256 and P-384 EDCHE keys as part of TLS secured connections to the TNMS server.

ASPP14:FCS_CKM.1/SK:

The TOE generates 128 and 256-bit AES keys during the TLS handshake and uses its Bouncy Castle cryptographic library to generate the random values used during the handshake. As required by ASPP14, the TOE's Bouncy Castle library calls the java.security.SecureRandom class [specifically calling SecureRandom.generateSeed()] to obtain a 256-bit seed, which is assumed to contain 256-bits of entropy.

ASPP14:FCS_CKM.2:

The TOE uses RSA-based key establishment and ECDHE as part of its TLS connection to the TNMS server

ASPP14:FCS_COP.1:

ASPP14:FCS_COP.1:

The TOE's Bouncy Castle cryptographic library (version 1.0.2.1) provides the following algorithm implementations and has algorithms certificates for each.

SFR	Algorithm	NIST Standard	Cert#
FCS_CKM.1/AK (Key Gen)	ECDH Key Generation P-256, 384, 521	FIPS 186-4, ECDSA	A2313
	RSA IFC key generation 2048 bits	FIPS 186-4, RSA	A2313
FCS_CKM.2 (Key Establishment)	ECDH Key Exchange	SP 800-56A, KAS ECC	A2313
	RSA Key Exchange	N/A	
FCS_COP.1/SKC	AES 128/256 CBC, GCM	FIPS 197, SP 800-38A/D	A2313
FCS_COP.1/Hash	SHA Hashing SHA-1, 256, 384, 512	FIPS 180-4	A2313
FCS_COP.1/Sig	RSA Sign/Verify 2048 bits	FIPS 186-4, RSA	A2313
	ECDSA Sign/Verify P-256, 384, 521	FIPS 186-4, ECDSA	A2313

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1/KeyedHash	HMAC-SHA HMAC-SHA 1, 256, 384, 512	FIPS 198-1 & 180-4	A2313
FCS_RBG_EXT.2 (Random)	DRBG Bit Generation Hash DRBG 256 bits	SP 800-90A	A2313

Table 6-1 Bouncy Castle CAVP Certificates

ASPP14:FCS_COP.1/Hash:

The TOE uses SHA-1, SHA-256, SHA-384, and SHA-512 hashing when verifying the TLS server's authentication signature. The TOE also uses SHA-1 when hashing the user's password (combined with a 64-bit random salt).

ASPP14:FCS_COP.1/KeyedHash:

The TOE uses HMAC as part of TLS (for encrypted data integrity).

ASPP14:FCS_COP.1/Sig:

The TOE verifies the server's digital signatures on its TLS certificate message.

ASPP14:FCS_COP.1/SKC:

The TOE can negotiate AES-CBC and AES-GCM as part of TLS connections. When generating a starting initial value (IV), the TOE generates a random number using its Bouncy Castle random number generator.

ASPP14:FCS_RBG_EXT.1:

The TOE implements a DRBG in its Bouncy Castle library and uses that DRBG when generating per-user salts as well as when generating random values used in TLS handshakes and PKBDF credential transformation. The TOE also invokes the platforms-provided DRBG functionality in order to obtain seeding material for its DRBG.

ASPP14:FCS_RBG_EXT.2:

The TOE obtains entropy to seed its DRBG from the platform through the BCryptGenRandom Windows API.

ASPP14:FCS_STO_EXT.1:

The TOE does not store any credentials.

PKGTLS11:FCS_TLSC_EXT.1:

The TOE supports the cipher suites selected in section 5.1.1.14 above, and the TOE uses the user provided server name (either a FQDN or an IP address) when checking the server's certificate for the expected reference identifier. The TOE supports Common Name, SAN:FQDN, SAN:IP reference identifiers, and wildcards and does not support certificate/public key pinning.

The TOE supports the ECDHE groups secp256r1, secp384r1, and secp521r1.

6.2 User data protection

ASPP14:FDP_DAR_EXT.1:

The TOE does not store any sensitive data. The TOE is a visual client for the TNMS Server, which the TNMS Server is the endpoint that processes all sensitive data. The TNMS Client stores basic information about its connection to the TNMS Server, however none of this is considered sensitive data.

ASPP14:FDP_DEC_EXT.1:

The TOE requires an active network connection.

ASPP14:FDP_NET_EXT.1:

The TOE allows a user to initiate a TLS secure network connection to the TNMS server.

6.3 Identification and authentication

ASPP14:FIA_X509_EXT.1:

The TOE performs certificate validity checking within its Bouncy Castle cryptographic library and examines each certificate in the path (starting with the peer's certificate) and first checks for validity of that certificate (e.g., has the certificate expired; or if not yet valid, whether the certificate contains the appropriate X.509 extensions [e.g., the CA flag in the basic constraints extension for a CA certificate, or that a server certificate contains the Server Authentication purpose in the ExtendedKeyUsagefield]), then verifies each certificate in the chain (applying the same rules as above, but also ensuring that the Issuer of each certificate matches the Subject in the next rung "up" in the chain and that the chain ends in a self-signed certificate present in either the TOE'S trusted anchor database or matches a specified Root CA), and finally the TOE performs revocation checking for all certificates in the chain. The TOE supports CRL as specified for RFC 5280 Section 6.3 for RSA certificates and CRL as specified in RFC 8603 for ECDSA certificates used in revocation checks under TLS connections.

ASPP14:FIA_X509_EXT.2:

The TOE validates X.509 certificates through outbound TLS connections against the administrator configured trusted anchor database. TLS server certificates that cannot be validated will not be accepted by the TOE and the trusted channel will not be established.

6.4 Security management

ASPP14:FMT_CFG_EXT.1:

The TOE requires no credential of its own. However, the TOE accepts a username and password, and the TOE forwards these to the TNMS server (through a TLS protected connection and after first hashing the user's password with SHA-1).

ASPP14:FMT_MEC_EXT.1:

The TOE makes use of its application storage area within the Windows filesystem to store the configured root CA.

ASPP14:FMT_SMF.1:

The TOE allows a user to configure (add, query, remove) the roots CA to which the TNMS server's peer certificate must chain and to initiate a connection to the server (after specifying their username, password, and the server's IP address or FQDN).

6.5 Privacy

ASPP14:FPR_ANO_EXT.1:

The TOE does not transmit any PII. The TOE only transmits the TNMS Server's username and hashed password.

6.6 Protection of the TSF

ASPP14:FPT_AEX_EXT.1:

The TOE does not specify any compilation flags when compiling its Java code; however, unlike native code, Java does not suffer from buffer overflow vulnerabilities. For any native executables bundled with the TOE, the corresponding compilation flags are used including /DYNAMICBASE, /GS, and /NXCOMPAT

ASPP14:FPT_API_EXT.1:

The TOE uses the following list of APIs:

java.security.AccessControlContext

java.security.AccessControlException

java.security.AccessController

java.security.cert.CertificateException

java.security.cert.CertificateFactory

java.security.cert.CRLException

java.security.cert.X509Certificate

java.security.cert.X509CRL

java.security.CodeSource

java.security.GeneralSecurityException

java.security.InvalidAlgorithmParameterException

java.security.InvalidKeyException

java.security.InvalidParameterException

java.security.Key

java.security.KeyManagementException

java.security.KeyStore

java.security.KeyStoreException

java.security.MessageDigest

java.security.NoSuchAlgorithmException

java.security.Permission

java.security.PermissionCollection

java.security.Permissions

java.security.Principal

java.security.PrivilegedAction

java.security.PrivilegedActionException

java.security.PrivilegedExceptionAction

java.security.ProtectionDomain

java.security.Provider

java.security.SecureRandom

java.security.Security

java.security.spec.AlgorithmParameterSpec

javax.security.auth.callback.Callback

javax.security.auth.callback.CallbackHandler

javax.security.auth.callback.NameCallback

javax.security.auth.callback.PasswordCallback

javax.security.auth.callback.UnsupportedCallbackException

javax.security.auth.kerberos.KerberosPrincipal

javax.security.auth.kerberos.KerberosTicket

javax.security.auth.login.AppConfigurationEntry.LoginModuleControlFlag

javax.security.auth.login.AppConfigurationEntry

javax.security.auth.login.Configuration

javax.security.auth.login.LoginContext

javax.security.auth.login.LoginException

javax.security.auth.Subject

ASPP14:FPT_IDV_EXT.1:

The TOE is versioned with a major and minor version number, as well as a build number that is incremented with every update to the TOE.

ASPP14:FPT_LIB_EXT.1:

ASPP14:FPT_TUD_EXT.1 / ALC_TSU_EXT.1:

The vendor packages updates to the TOE in an EXE installer format (as opposed to being bundled with the Windows platform itself) and relies upon the Windows 10 operating system to verify the installation package's signature before installing. Updates are signed by Infinera's Window developer key and display under the Windows platform with the Name of Signer as www.infinera.com. Updates are obtained through Infinera's Customer Service Portal (https://support.infinera.com/) and are installed using the same process as the initial installation.

The vendor provides timely security updates for the TOE in case vulnerabilities have been discovered. Reported vulnerabilities and defects are investigated and rated based on the threat and result of the impact analysis and then scheduled for an upcoming bug fix release based on the severity. The vendor aims for security updates as soon as possible with a maximum of 30 days. Third party library updates are also included as a part of the TOE's update. The vendor actively monitors both internal and third-party components and accepts vulnerability reports through the Infinera's Customer portal (https://www.support.infinera.com/).

6.7 Trusted path/channels

ASPP14:FTP_DIT_EXT.1:

The TOE encrypts all transmitted data using TLS.