# VMware NSX-T Data Center 3.1 Security Target

Version 1.6
July 12, 2022

VMware, Inc
3401 Hillview Avenue, Palo Alto, CA 94304

# Abstract

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), VMware NSX-T Data Center 3.1. This Security Target (ST) is conformant to the requirement of Collaborative Protection Profile for Network Devices (or NDcPP) v2.2e.

# References

| | |
|---|---|
| [CC1] | Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 5, April 2017 |
| [CC2] | Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 5, April 2017: Part 2 extended |
| [CC3] | Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 5, April 2017: Part 3 conformant |
| [CC_Add] | CC and CEM Addenda, Exact Conformance, Selection-Based SFRs, Optional SFRs, CCDB-2017-05-xxx, Version 0.5, May 2017 |

# Product Guide Reference

| |
|---|
| VMware NSX-T Data Center 3.1 Common Criteria Evaluated Configuration Guide AGD |
| VMware NSX-T Data Center 3.1 Release Note |
| VMware NSX-T Data Center 3.1 Installation Guide |
| VMware NSX-T Data Center 3.1 Upgrade Guide |
| VMware NSX-T Data Center 3.1 Administration Guide |

# Table of Content

# Table of Tables

# Table of Figures

# 1 Introduction

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

## 1.1 ST Reference

| | |
|---|---|
| ST Title | VMware NSX-T Data Center 3.1 Security Target |
| ST Revision | 1.6 |
| ST Draft Date | July 12, 2022 |
| Author | VMware, Inc. |
| cPP Conformance | NDcPP v2.2e |

## 1.2 TOE Reference

| | |
|---|---|
| TOE Title | VMware NSX-T Data Center 3.1 |
| TOE Version | 3.1.3 |

**Table 1 – TOE Reference**

## 1.3 About this document

This Security Target follows the following format:

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

**Table 2 – Document Organization**

## 1.4 TOE Overview

The Target of Evaluation (TOE) is VMware NSX-T Data Center 3.1, software-only network virtualization platform that programmatically provisions and manages virtual networks through software network devices and provides network overlay for virtual environments.

In much the same way the server virtualization programmatically creates, snapshots, deletes and restores software-based virtual machines (VMs), NSX-T network virtualization programmatically creates, and deletes software-based virtual networks.

With network virtualization, NSX-T reproduces Layer 2 through Layer 7 networking services (for example, bridging, switching, routing, firewalling, loadbalancer) in software. As a result, these services can be programmatically assembled to produce unique, isolated virtual networks in a matter of seconds.

VMware NSX-T Data Center 3.1 works by implementing three separate but integrated planes:

- The management plane, via NSX-T REST API, provides a single API entry point to the system. It is responsible for maintaining user configuration, handling user queries, and performing operational tasks on all management, control, and data plane components. This part resides within NSX-T Unified Appliance, which is part of the TOE.
- The control plane computes the runtime state of the system based on configuration from the management plane. It is also responsible for disseminating topology information reported by the data plane elements and pushing configuration to forwarding engines. This part resides within NSX-T Unified Appliance, which is part of the TOE.
- The data plane performs forwarding or transformation of packets. This data plane is the NSX-T Edge appliance, which is part of the TOE, however the data plane functions and control plane functions are excluded from the evaluated functionality. The data traffic flows through the node is from the virtual NICs form the Virtual Machines running on ESXi.

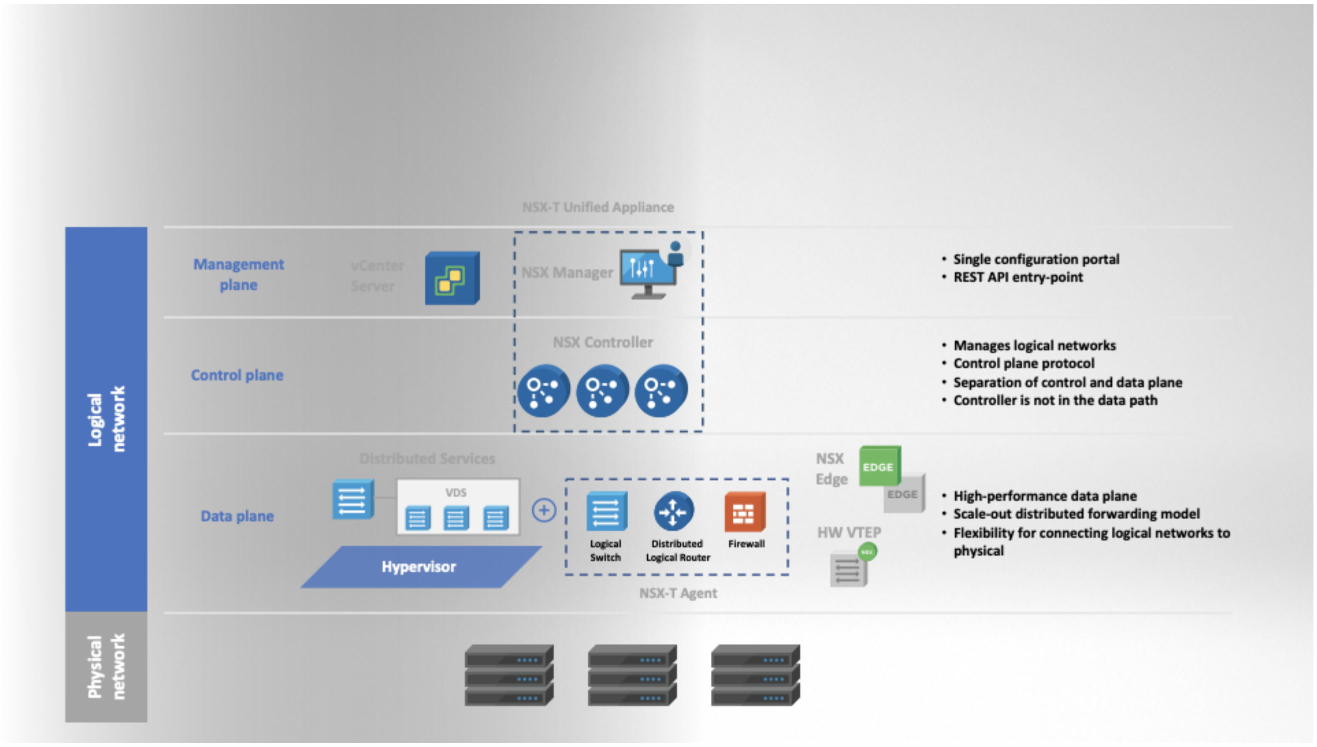
NSX-T Unified Appliance
Logical network
Physical network
Management plane
vCenter Server
NSX Manager
Control plane
NSX Controller
Data plane
Distributed Services
VDS
Hypervisor
Logical Switch
Distributed Logical Router
Firewall
NSX-T Agent
NSX Edge
EDGE
EDGE
HW VTEP
NSX
• Single configuration portal
• REST API entry-point
• Manages logical networks
• Control plane protocol
• Separation of control and data plane
• Controller is not in the data path
• High-performance data plane
• Scale-out distributed forwarding model
• Flexibility for connecting logical networks to physical

**Figure 1 – Product Overview**

## 1.5 TOE Description

### 1.5.1 TOE Description Overview

The Target of Evaluation (TOE) is VMware NSX-T Data Center 3.1, a VMware network device software product that provides and manages virtual networking components. The TOE is designed as a network virtualization platform, providing the ability to implement and virtualize networks across multiple ESXi nodes and virtual machines (VMs) while providing isolation and efficient use of network resources. This allows for implementation of wide array of various data center workloads or on-demand infrastructure.

For the purpose of testing of the identified TOE, the evaluated TOE configuration is as followings: VMware NSX-T Data Center 3.1 on hypervisor VMware ESXi 6.7 running Ubuntu 18.04 on Dell Power Edge R740 with Intel Xeon Gold 6230R (Cascade Lake).

The TOE provides functionality to enforce and support auditing, cryptographic operations, network separation, encrypted channels, identification/authentication, security management, and protection of the TSF. Administrators can configure virtual network.

In VMware's network virtualization solution, the following components are the essential building blocks that make up the virtualized computing environment:

- NSX-T Unified Appliance is a virtual appliance configured to run NSX-T application roles (Manager, Policy, and Controller).
- The NSX-T Edge is a virtual appliance that provides routing services and connectivity to networks that are external to the NSX-T deployment.

The components described above make a basic virtualized environment ready for virtualized networking. The NSX-T Unified Appliance provides a single API entry point to the system, persists user configuration, handles user queries, and performs operational tasks. The TOE is configured for a single instance of ESXi 6.7 Hypervisor, which includes a single NSX-T Unified Appliance instance, and a single instance of the NSX-T Edge appliance.

TOE supports both local and remote administration. The TOE provides a local console interface which supports CLI. REST API over TLS is the remote interface.

### 1.5.2 Physical Boundary

VMware NSX-T Data Center 3.1, the TOE, is network virtualization software that provides network functionality for ESXi hypervisors and the virtual machines running on those hypervisors. The TOE abstracts network functions, including bridging, switching, and routing to create a virtual network that can connect to the physical network.

The TOE satisfies Case 1 as depicted in NDcPP v2.2e. The TOE is represented by the vND alone.
The following figure shows the detailed TOE boundary of VMware NSX-T Data Center 3.1. The different subsystems and interfaces are described briefly below:

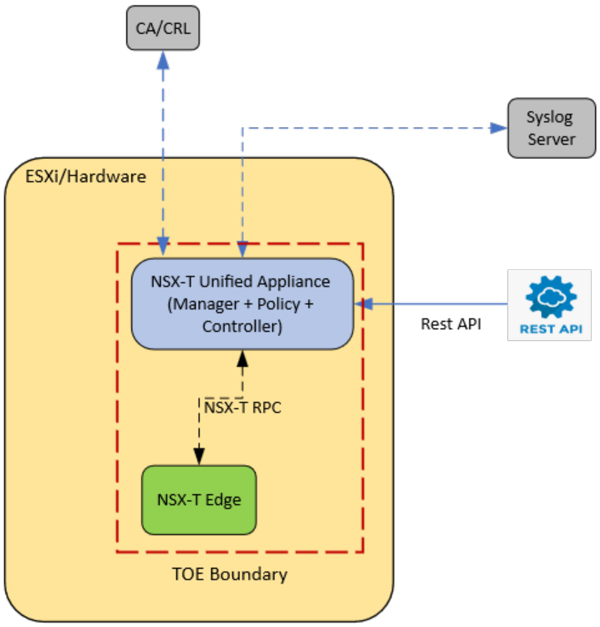

**Figure 2 - TOE Physical Boundary**[1]

**NSX-T Unified Appliance:**

The NSX-T Unified Appliance contains a set of applications and daemons which implement the management functionality of NSX-T. Applications roles include Manager, Policy and Controller; and includes the HTTPS reverse proxy which serves as a TLS endpoint for the REST API. Policy controls the desired state configurations. Manager manages persistent data and communication between application roles; and communicates with Edge using TLS 1.1/1.2. Controller translates desired state into realized state and communicates with NSX-T Edge using TLS 1.1/1.2.

NSX Unified Appliance (Manager + Policy + Controller) is installed as a Virtual Machine (VM) on the ESXi hypervisor.

**NSX-T Edge:**

The NSX-T Edge Virtual Appliance contains a set of applications and daemons which bridge physical networking to the virtual network maintained by NSX-T.

**Syslog Server:**

The TOE establishes a trusted channel communication with the syslog server using TLS 1.1/1.2.

**CA Server:**

Server which contains the updated revocation list for the TOE.

**REST API:**

The main interface to NSX-T functionality is via a RESTful interface using HTTP over TLS 1.1/1.2. The REST API supports authentication using session-based authentication using a username and password. The API exposes interfaces to configure virtual networking constructs, and to observe the operational status of those constructs. Additionally, the API allows you to observe the operational status of the physical underlay network elements.

[1] Note: The data plane functions are excluded from the evaluated functionality of the TOE. In addition, the internal NSX-T communication path is internal to the TOE. The TLS channel which provides the NSX-T is also excluded from the evaluated functionality.

### 1.5.3 Logical Boundary

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

### *1.5.3.1 Logical Boundary Rationale for Security Audit (FAU)*

| Security Function | Description |
|---|---|
| Security Audit (FAU) | The TOE generates audit records for all security-relevant events. For each audited events, the TOE records the date and time, the type of event, the subject identity, and the outcome of the event. The resulting records are stored on Unified Appliance and can be sent securely to a designated log server for archiving. Security Administrators, using the appropriate REST API commands, can also view audit records locally. The TOE provides a reliable timestamp relying on the appliance's to built-in clock. |

### *1.5.3.2 Logical Boundary Rationale for Cryptographic Support (FCS)*

| Security Function | Description |
|---|---|
| Cryptographic Support (FCS) | The TOE provides cryptography support for secure communications and protection of information. The cryptographic services provided by the TOE are listed in Table 3 - TOE Provided Cryptography below. The TOE implements the secure protocols - TLS/HTTPS on the server side and TLS on the client side. The TOE implements deletion procedures to mitigate the possibility of disclosure or modification of CSPs.<br>The TOE uses two types of dedicated cryptographic modules to manage CSPs: VMware BC-FJA (Bouncy Castle FIPS Java API) module for Java based implementations of TLS/HTTPS, key stores, and trust stores; and VMware's OpenSSL FIPS Object module for TLS/HTTPS, key stores, and trust stores. The algorithm certificate references are listed in the tables below (Table 4 – VMware's OpenSSL FIPS Object Module Algorithm and Table 5 – VMware BC-FJA (Bouncy Castle FIPS Java API) Module Algorithm). |

The following table lists all cryptography provided within TOE:

| Cryptographic Method | Usage within the TOE |
|---|---|
| TLS Establishment | Used to establish initial TLS session |
| ECDH Key Agreement | Used in TLS session establishment |
| RSA Key Generation | Used to create key-pairs and X.509 certificates for use in TLS protocols |
| RSA Signature Services | Used in TLS session establishment.<br>Used in secure software update |
| SP 800-90 DRBG | Used in TLS session establishment |
| SHS | Used in secure software update |
| HMAC-SHS | Used to provide TLS traffic integrity verification |
| AES | Used to encrypt TLS traffic |

**Table 3 – TOE Provided Cryptography**

Algorithms under VMware's OpenSSL FIPS Object Module cryptography module are listed in the table:

| Algorithm | Description | Mode Supported | CAVP Cert. # | Standards | Operational Environment |
|---|---|---|---|---|---|
| AES | Used for symmetric encryption/decryption | GCM (128 and 256 bits)<br>CBC (128 and 256 bits) | A1292 | SP 800-38D<br>SP 800-38A | Ubuntu 18.04 on ESXi 6.7 with Intel Xeon Gold 6230R (Cascade Lake) with AES-NI |
| SHS (SHA) | Cryptographic hashing services | Byte Oriented<br>SHA-1,<br>SHA-256,<br>SHA-384 | A1292 | FIPS 180-4 | |
| HMAC | Keyed hashing services and software integrity test | Byte Oriented<br>HMAC-SHA-1,<br>HMAC-SHA-256,<br>HMAC-SHA-384 | A1292 | FIPS 198 | |
| DRBG | Deterministic random bit generation services in accordance with ISO/IEC 18031:2011 | Hash_DRBG (512)<br>CTR_DRBG (AES-256) | A1292 | SP 800-90A | |
| RSA | Signature Generation and Verification | FIPS PUB 186-4 Key Generation (2048-bit, 3072 bit key) | A1292 | FIPS 186-4 | |
| CVL – KAS-ECC | Key Agreement | NIST Special PUB 800-56A | A1292 | SP 800-56Ar3 | |

**Table 4 – VMware's OpenSSL FIPS Object Module Algorithm**

| Algorithm | Description | Mode Supported | CAVP Cert. # | Standards | Operational Environment |
|---|---|---|---|---|---|
| AES | Used for symmetric encryption/decryption | GCM (128 and 256 bits)<br>CBC (128 and 256 bits) | C2174 | SP 800-38D<br>SP 800-38A | Ubuntu 18.04 with JDK 8 on VMware ESXi 6.7 on Intel Xeon Gold 6230R (Cascade Lake) |
| SHS (SHA) | Cryptographic hashing services | Byte Oriented<br>SHA-1,<br>SHA-256,<br>SHA-384 | C2174 | FIPS 180-4 | |
| HMAC | Keyed hashing services and software integrity test | Byte Oriented<br>HMAC-SHA-1,<br>HMAC-SHA-256,<br>HMAC-SHA-384 | C2174 | FIPS 198 | |
| DRBG | Deterministic random bit generation services in accordance with ISO/IEC 18031:2011 | Hash_DRBG (512)<br>CTR_DRBG (AES-256) | C2174 | SP 800-90A | |
| RSA | Signature Generation and Verification | FIPS PUB 186-4 Key Generation (2048-bit, 3072 bit key) | C2174 | FIPS 186-4 | |
| CVL – KAS-ECC | Key Agreement | NIST Special Publication 800-56A | C2174 | SP 800-56Ar3 | |

**Table 5 – VMware BC-FJA (Bouncy Castle FIPS Java API) Module Algorithm**

#### 1.5.3.3 Logical Boundary Rationale for Identification and Authentication (FIA)

| Security Function | Description |
|---|---|
| Identification and Authentication (FIA) | Security Administrators are identified and authenticated prior to being allowed access to any of the services other than the display of the warning banner. The REST API requires user name and password for authentication. The identification and authentication credentials are confirmed against a local user database. The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS/HTTPS connections.<br>The TOE provides the capability to set password minimum length rules to ensure the use of strong passwords in attempts to protect against brute force attacks. The TOE also accepts passwords composed of a variety of characters to support complex password composition. During authentication, no indication is given of the characters composing the password. |

#### 1.5.3.4 Logical Boundary Rationale for Security Management (FMT)

| Security Function | Description |
|---|---|
| Security Management (FMT) | The TOE provides secure administrative services for management of general TOE configuration and TOE security functionality. There are two types of administrative users within the system: Security Administrator and Auditor (read only). All of the management functions are restricted to Security Administrators. The TOE administration occurs through REST API. The TOE provides the ability to perform the following actions:<br>• Administer the TOE locally and remotely<br>• Configure the access banner<br>• Configure the cryptographic services<br>• Update the TOE and verify the updates using digital signature capability prior to installing those updates<br>• Specify the time limits of session inactivity |

#### 1.5.3.5 Logical Boundary Rationale for Protection of the TSF (FPT)

| Security Function | Description |
|---|---|
| Protection of the TSF (FPT) | The TOE implements a number of measures to protect the integrity of its security features:<br>• The TOE protects CSPs, including stored passwords and cryptographic keys, so they are not directly viewable or accessible in plaintext.<br>• The TOE ensures that reliable time information is available for log accountability. The time can be configured through the REST API.<br>The TOE performs self-tests to detect internal failures and protect itself from malicious updates. |

#### 1.5.3.6 Logical Boundary Rationale for TOE Access (FTA)

| Security Function | Description |
|---|---|
| TOE Access (FTA) | The TOE will display a customizable banner when an administrator initiates a session. The TOE also enforces an administrator-defined inactivity timeout after which any inactive session is automatically terminated. Once a session has been terminated, the TOE requires the user to re-authenticate. |

#### *1.5.3.7 Logical Boundary Rationale for Trusted Path/Channels (FTP)*

| **Security Function** | **Description** |
|---|---|
| Trusted Path/Channels (FTP) | The TOE establishes a trusted path between the Unified Appliance and the administrative REST API using TLS/HTTPS. The TOE establishes a secure connection using TLS for:<br>• Sending syslog data to a log server. |

### 1.5.4 Non-TOE Hardware/Software/Firmware

The following components are not within the TOE Boundary and are located in the TOE environment:

- Syslog (Audit) Server (rsyslogd 8.20 was used in the evaluated configuration)
- VMware ESXi 6.7
- NSX Agent software (installed on the ESXi hypervisor)
- NSX and vSphere Distributed Switches (NVDS/VDS)
- Certificate Authority Server (CA) (XCA 2.1.0 was used in the evaluated configuration)

### 1.5.5 Summary of out-of-scope items

The TOE supports a number of features that are not part of the core functionality. These features are not included in the scope of the evaluation:

- Unified Appliance clustering is not restricted; however, it is not evaluated.
- Any integration and/or communication with authentication servers such as vIDM is not evaluated.
- The use of SNMPv3 for monitoring is not restricted; however, it is not evaluated.
- Synchronization with an external NTP server is not restricted; however, this functionality is not evaluated.
- The Log Insight interface (an alternative Audit Server) is not enabled by default and is not evaluated.
- CLI (using SSH communications) is not enabled by default and is not evaluated.
- The TOE’s debug mode is not intended for normal use and is not evaluated.
- Public and Hybrid Cloud functionality is not enabled by default; and is not evaluated.
- Container functionality is not enabled by default; and is not evaluated.
- The intra-TOE TLS connection between the UA and the NSX-T Edge is not evaluated and an Edge platform residing on another ESXi is not evaluated.
- NSX-T Edge Data-plane Services (or NVDS/VDS) and NSX Agents are excluded from the TOE; and are not evaluated.
- vCenter Server

# 2 Conformance Claim

## 2.1 CC Conformance Claim

This TOE is conformant to:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 5, April 2017: Part 2 extended
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 5, April 2017: Part 3 conformant

This ST provides exact conformance to the items listed in the previous section. The security problem definition, security objectives, and security requirements in this ST are all taken from the Protection Profile (PP), performing only the operations defined there.

## 2.2 PP Conformance Claim

This TOE is conformant to:

- collaborative Protection Profile for Network Devices, Version 2.2e, 23 March 2020 [NDcPPv2.2e].

## 2.3 Technical Decisions

| Technical Decision | Applicable (Yes/No) | Exclusion Rationale (If Applicable) |
|---|---|---|
| TD0636 - NIT Technical Decision for Clarification of Public Key User Authentication for SSH | No | SSH is not claimed. |
| TD0635 - NIT Technical Decision for TLS Server and Key Agreement Parameters | Yes | |
| TD0634 - NIT Technical Decision for Clarification required for testing IPv6 | No | IP identifiers are not supported. |
| TD0633 - NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance | No | IPSec is not claimed. |
| TD0632 - NIT Technical Decision for Consistency with Time Data for vNDs | Yes | |
| TD0631 - NIT Technical Decision for Clarification of public key authentication for SSH Server | No | SSH is not claimed. |
| TD0592 – NIT Technical Decision for Local Storage of Audit Records | Yes | |
| TD0591 – NIT Technical Decision for Virtual TOEs and hypervisors | Yes | |
| TD0581 – NIT Technical Decision for Elliptical curve-based key establishment and NIST SP 800-56Arev3 | Yes | |
| TD0580 – NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e | No | DH14 is not used and hence not claimed |
| TD0572 – Restricting FTP_ITC.1 to only IP address identifiers | Yes | |
| TD0571 – Guidance on how to handle FIA_AFL.1 | Yes | |
| TD0570 – Clarification about FIA_AFL.1 | Yes | |
| TD0569 – Session ID Usage Conflict in FCS_DTLSS_EXT.1.7 | No | DTLSS is not claimed |
| TD0564 – Vulnerability Analysis Search Criteria | Yes | |
| TD0563 – Clarification of audit date information | Yes | |
| TD0556 – NIT Technical Decision for RFC 5077 question | Yes | |

| | | |
|---|---|---|
| TD0555 – NIT Technical Decision for RFC Reference incorrect in TLSS Test | Yes | |
| TD0547 – Clarification on developer disclosure of software components as part of AVA_VAN | Yes | |
| TD0546 – DTLS - clarification of Application Note 63 | No | DTLS is not claimed |
| TD0538 – Outdated link to allowed-with list | Yes | |
| TD0537 – Incorrect reference to FCS_TLSC_EXT.2.3 | Yes | |
| TD0536 – Update Verification Inconsistency | Yes | |
| TD0528 – Missing EAs for FCS_NTP_EXT.1.4 | No | NTP is not claimed |
| TD0527 – Updates to Certificate Revocation Testing (FIA_X509_EXT.1 | No | ECDSA is not used in X.509 certificates |

**Table 6 – Technical Decisions**

# 3 Security Problem Definition

This Security Target provides exact conformance to the Protection Profile(s) described in the conformance claims above. The security problem definition, security objectives and security requirements in this Security Target are all taken from the applicable Protection Profile(s) performing only operations defined there.
As this TOE is not distributed, none of the threats/assumptions/OSPs relating to distributed TOEs are specified for this TOE.

## 3.1 Threats

The followings are threats for this TOE as defined in NDcPP v2.2e Section 4.1.

### 3.1.1 Communications with the Network Device

#### *3.1.1.1 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.

#### *3.1.1.2 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.

#### *3.1.1.3 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.

#### *3.1.1.4 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.

### 3.1.2 Valid Updates

#### *3.1.2.1 T.UPDATE_COMPROMISE*

Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.

### 3.1.3 Audited Activity

#### *3.1.3.1 T.UNDETECTED_ACTIVITY*

Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.

### 3.1.4 Administrator and Device Credentials and Data

#### *3.1.4.1 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.

#### *3.1.4.2 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.

### 3.1.5 Device Failure

#### *3.1.5.1 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

The followings are assumptions made for this TOE are as defined in NDcPP v2.2e Section 4.2.

### 3.2.1 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 unauthorized 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.

### 3.2.2 A.LIMITED_FUNCTIONALITY

The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality).

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.

### 3.2.3 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).

### 3.2.4 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).

### 3.2.5 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.

### 3.2.6 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.

### 3.2.7 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.

### 3.2.8 A.VS_TRUSTED_ADMINISTRATOR (applies to vNDs only)

The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.

### 3.2.9 A.VS_REGULAR_UPDATES (applies to vNDs only)

The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

### 3.2.10 A.VS_ISOLATON (applies to vNDs only)

For vNDs, it is assumed that the VS provides, and is configured to provide sufficient isolation between software running in VMs on the same physical platform. Furthermore, it is assumed that the VS adequately protects itself from software running inside VMs on the same physical platform.

### 3.2.11 A.VS_CORRECT_CONFIGURATION (applies to vNDs only)

For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.

## 3.3 Organizational Security Policies

The followings are OSP applied for this TOE is as defined in NDcPP Section 4.3. No additional OSPs are identified and no modification to the statement of OSPs is made for this TOE.

### 3.3.1 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

This TOE is not distributed; therefore, none of the objectives relating to distributed TOEs are specified for this TOE.

## 4.1 Security Objectives for the Operational Environment

Security objectives for the operational environment of this TOE as defined in NDcPP v2.2e Section 5.1 is as followings:

### 4.1.1 OE.PHYSICAL

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

### 4.1.2 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.

### 4.1.3 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.

### 4.1.4 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.

### 4.1.5 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.

### 4.1.6 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.

### 4.1.7 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.

### 4.1.8 OE.VM_CONFIGURATION (applies to vNDs only)

For vNDs, the Security Administrator ensures that the VS and VMs are configured to

- reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and
- correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting).

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

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

# 5 Security Functional Requirements

All security functional requirements are taken from the NDcPP v2.2e. The SFRs are presented in accordance with the conventions described in NDcPP v2.2e Section 6.1, and section "5.1 Document Convention" of this document.

## 5.1 Document Convention

- This document follows the same conventions as those applied in NDcPP 2.2e in the completion of operations on Security Functional Requirements as outlined below:
    - Selection: indicated with underlined text
    - Refinement made in the PP: the refinement text is indicated with **bold text** and ~~strikethroughs~~;
    - Assignment: indicated with *italicized text*;
    - Assignment within a selection: indicated with *italicized and underlined text*
    - Iteration: indicated by adding a string starting with "/" (e.g. "FCS_COP.1/Hash").
    - Unaltered SFRs are stated in the form used in [CC2] or their extended component definition (ECD);
- Extended SFRs are identified by having a label "EXT" at the end of the SFR name.
- Where compliance to RFCs is referred to in SFRs, this is intended to be demonstrated by completing the corresponding evaluation activities in [SD] for the relevant SFR.

## 5.2 Security Audit (FAU)

### 5.2.1 Security Audit Data Generation (FAU_GEN)

### *5.2.1.1 FAU_GEN.1 Audit data generation*

| FAU_GEN.1 | Audit Data Generation |
|---|---|

#### *5.2.1.1.1 FAU_GEN.1.1*

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

a) Start-up and shut-down of the audit functions;

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

c) *All administrative actions comprising:*

- *Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).*
- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *Resetting passwords (name of related user account shall be logged).*
- *[no other actions];*

d) *Specifically defined auditable events listed in* Table 7 - FAU_GEN Auditable Events.

#### *5.2.1.1.2 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 7 - FAU_GEN Auditable 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_RBG_EXT.1 | None. | None. |
| FCS_TLSC_EXT.1 | Failure to establish a TLS Session | Reason for failure |
| FCS_TLSC_EXT.2 | None | None |
| 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). |
| FIA_UAU.7 | None. | None. |
| FIA_X509_EXT.1/Rev | • Unsuccessful attempt to validate a certificate<br>• Any addition, replacement or removal of trust anchors in the TOE's trust store | • Reason for failure of certificate validation<br>• 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/Services | None. | 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. |

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FPT_STM_EXT.1 | Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1) | For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address). |
| FTA_SSL_EXT.1 (if “terminate the session” is selected) | The termination of a local 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.<br>• Termination of the trusted channel.<br>• 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.<br>• Termination of the trusted path.<br>• Failure of the trusted path functions. | None. |

**Table 7 - FAU_GEN Auditable Event**

#### *5.2.1.2 FAU_GEN.2 User identity association*

| FAU_GEN.2 | User identity association |
|---|---|

##### *5.2.1.2.1 FAU_GEN.2.1*

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

### 5.2.2 Security audit event storage (Extended – FAU_STG_EXT)

#### *5.2.2.1 FAU_ STG_EXT.1 Protected Audit Event Storage*

| FAU_STG_EXT.1 | Protected Audit Event Storage |
|---|---|

##### *5.2.2.1.1 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.

##### *5.2.2.1.2 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,*

].

##### *5.2.2.1.3 FAU_STG_EXT.1.3*

The TSF shall [*drop new audit data*] when the local storage space for audit data is full.

## 5.3 Cryptographic Support (FCS)

### 5.3.1 Cryptographic Key Management (FCS_CKM)

#### *5.3.1.1 FCS_CKM.1 Cryptographic Key Generation (Refinement)*

| | |
|---|---|
| **FCS_CKM.1** | **Cryptographic Key Generation** |

##### *5.3.1.1.1 FCS_CKM.1.1*

The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm: [

- *RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;*
- *ECC schemes using 'NIST curves' [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;*

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

#### *5.3.1.2 FCS_CKM.2 Cryptographic Key Establishment (Refinement)*

| | |
|---|---|
| **FCS_CKM.2** | **Cryptographic Key Establishment** |

##### *5.3.1.2.1 FCS_CKM.2.1*

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

- *RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1";*
- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";*

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

#### *5.3.1.3 FCS_CKM.4 Cryptographic Key Destruction*

| | |
|---|---|
| **FCS_CKM.4** | **Cryptographic Key Destruction** |

##### *5.3.1.3.1 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], destruction of reference to the key directly followed by a request for garbage collection];*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [*
    - *logically addresses the storage location of the key and performs a [single-pass] overwrite consisting of [zeroes]]]*

that meets the following: *No Standard*.

### 5.3.2 Cryptographic Operation (FCS_COP)

#### *5.3.2.1 FCS_COP.1 Cryptographic Operation*

| **FCS_COP.1/DataEncryption** | **Cryptographic Operation (AES Data Encryption/ Decryption)** |
|---|---|

#### 5.3.2.1.1 *FCS_COP.1.1/DataEncryption*

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

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

#### 5.3.2.1.2 *FCS_COP.1.1/SigGen*

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

- *RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits, 3072 bits],*

]

that meet the following: [

- *For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,*

].

| **FCS_COP.1/Hash** | **Cryptographic Operation (Hash Algorithm)** |
|---|---|

#### 5.3.2.1.3 *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*] ~~and cryptographic key sizes [*assignment: cryptographic key sizes*]~~ and **message digest sizes [*160, 256, 384*] bits** that meet the following: ISO/IEC 10118-3:2004.

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

#### 5.3.2.1.4 *FCS_COP.1.1/KeyedHash*

The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384*] and cryptographic key sizes *[160 bits, 128 bits, 256 bits]* **and message digest sizes [*160, 256, 384*] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

## 5.3.3 Random Bit Generation (Extended – FCS_RBG_EXT)

### 5.3.3.1 *FCS_RBG_EXT.1 Random Bit Generation*

| **FCS_RBG_EXT.1** | **Random Bit Generation** |
|---|---|

#### 5.3.3.1.1 *FCS_RBG_EXT.1.1*

The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [*Hash_DRBG (512), CTR_DRBG (AES-256)*].

***ST Application Note:***

*TOE uses two different security modules for DRBG: OpenSSL uses CTR_DRBG (AES-256), BouncyCastle uses Hash_DRBG (512).*

#### 5.3.3.1.2 *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], [one hardware-based noise source]*] with a minimum of [*256 bits*] of entropy at least equal to the greatest security strength (according to ISO/IEC 18031:2011 Table C.1 “Security Strength Table for Hash Functions”), of the keys and hashes that it will generate.

## 5.3.4 Cryptographic Protocols (Extended – FCS_TLSC_EXT and FCS_TLSS_EXT Protocol)

### *5.3.4.1 FCS_TLSC_EXT.1 TLS Client Protocol Without Mutual Authentication*

| **FCS_TLSC_EXT.1** | **TLS Client Protocol Without Mutual Authentication** |
|---|---|

#### *5.3.4.1.1 FCS_TLSC_EXT.1.1*

The TSF shall implement [*TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)*] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[

- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*

]

*and no other ciphersuites.*

#### *5.3.4.1.2 FCS_TLSC_EXT.1.2*

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

#### *5.3.4.1.3 FCS_TLSC_EXT.1.3*

When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [

- *Not implement any administrator override mechanism*

].

#### *5.3.4.1.4 FCS_TLSC_EXT.1.4*

The TSF shall [*present the Supported Elliptic Curves/Supported Groups Extension with the following curves/groups: [secp256r1, secp384r1, secp521r1] and no other curves/groups*] in the Client Hello.

### *5.3.4.2 FCS_TLSC_EXT.2 TLS Client Support for Mutual Authentication*

| **FCS_TLSC_EXT.2** | **TLS Client Support for Mutual Authentication** |
|---|---|

#### *5.3.4.2.1 FCS_TLSC_EXT.2.1*

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

### *5.3.4.3 FCS_TLSS_EXT.1 TLS Server Protocol Without Mutual Authentication*

| **FCS_TLSS_EXT.1** | **TLS Server Protocol Without Mutual Authentication** |
|---|---|

##### *5.3.4.3.1 FCS_TLSS_EXT.1.1*

The TSF shall implement [*TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)*] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[

- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*

]

*and no other ciphersuites.*

##### *5.3.4.3.2 FCS_TLSS_EXT.1.2*

The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [*none*].

##### *5.3.4.3.3 FCS_TLSS_EXT.1.3*

The TSF shall perform key establishment for TLS using [*RSA with key size* [*2048 bits, 3072 bits*], *ECDHE curves* [*secp256r1, secp384r1, secp521r1*] *and no other curves*]].

##### *5.3.4.3.4 FCS_TLSS_EXT.1.4*

The TSF shall support [*no session resumption or session tickets*].

## 5.4 Identification and Authentication (FIA)

### 5.4.1 Authentication Failure Management (FIA_AFL)

#### *5.4.1.1 FIA_AFL.1 Authentication Failure Management (Refinement)*

| **FIA_AFL.1** | **Authentication Failure Management** |
|---|---|

##### *5.4.1.1.1 FIA_AFL.1.1*

The TSF shall detect when an Administrator configurable positive integer within [*1-5*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

##### *5.4.1.1.2 FIA_AFL.1.2*

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

### 5.4.2 Password Management (Extended – FIA_PMG_EXT)

#### *5.4.2.1 FIA_PMG_EXT.1 Password Management*

| | |
|---|---|
| **FIA_PMG_EXT.1** | **Password Management** |

##### 5.4.2.1.1 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 [*8*] and [*20*] characters.

### 5.4.3 User Identification and Authentication (Extended – FIA_UIA_EXT)

#### 5.4.3.1 FIA_UIA_EXT.1 User Identification and Authentication

| | |
|---|---|
| **FIA_UIA_EXT.1** | **User Identification and Authentication** |

##### 5.4.3.1.1 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;
- [*obtain non-security related static data for UI purposes*];

##### 5.4.3.1.2 FIA_UIA_EXT.1.2

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

### 5.4.4 User authentication (FIA_UAU) (Extended – FIA_UAU_EXT)

#### 5.4.4.1 FIA_UAU_EXT.2 Password-based Authentication Mechanism

| | |
|---|---|
| **FIA_UAU_EXT.2** | **Password-based Authentication Mechanism** |

##### 5.4.4.1.1 FIA_UAU_EXT.2.1

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

#### 5.4.4.2 FIA_UAU.7 Protected Authentication Feedback

| | |
|---|---|
| **FIA_UAU.7** | **Protected Authentication Feedback** |

##### 5.4.4.2.1 FIA_UAU.7.1

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

### 5.4.5 Authentication using X.509 certificates (Extended – FIA_X509_EXT)

#### 5.4.5.1 FIA_X509_EXT.1 X.509 Certificate Validation

| | |
|---|---|
| **FIA_X509_EXT.1/Rev** | **X.509 Certificate Validation** |

##### 5.4.5.1.1 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 [*Certificate Revocation List (CRL) as specified in RFC 5759 Section 5*]
- 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.*

#### 5.4.5.1.2 FIA_X509_EXT.1.2/Rev

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

### 5.4.5.2 FIA_X509_EXT.2 X509 Certificate Authentication

| **FIA_X509_EXT.2** | **X.509 Certificate Authentication** |
|---|---|

#### 5.4.5.2.1 FIA_X509_EXT.2.1

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

#### 5.4.5.2.2 FIA_X509_EXT.2.2

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

### 5.4.5.3 FIA_X509_EXT.3 X.509 Certificate Requests

| **FIA_X509_EXT.3** | **X.509 Certificate Requests** |
|---|---|

#### 5.4.5.3.1 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*].

***ST Application Note:***
*TOE supports multiple types of CSR requires, some do not contain device-specific-information.*

#### 5.4.5.3.2 FIA_X509_EXT.3.2

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

## 5.5 Security Management (FMT)

### 5.5.1 Management of functions in TSF (FMT_MOF)

#### 5.5.1.1 FMT_MOF.1/ManualUpdate Management of Security Functions Behavior

| FMT_MOF.1/ManualUpdate | Management of Security Functions Behaviour |
|---|---|

##### 5.5.1.1.1 FMT_MOF.1.1/ManualUpdate

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

#### 5.5.1.2 FMT_MOF.1/Services Management of security functions behavior

| FMT_MOF.1/Services | Management of Security Functions Behaviour |
|---|---|

##### 5.5.1.2.1 FMT_MOF.1.1/Services

The TSF shall restrict the ability to **start and stop** ~~the functions~~ **services** to *Security Administrators.*

#### 5.5.1.3 FMT_MOF.1/Functions Management of security functions behavior

| FMT_MOF.1/Functions | Management of Security Functions Behaviour |
|---|---|

##### 5.5.1.3.1 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*.

### 5.5.2 Management of TSF Data (FMT_MTD)

#### 5.5.2.1 FMT_MTD.1/CoreData Management of TSF Data

| FMT_MTD.1/CoreData | Management of TSF Data |
|---|---|

##### 5.5.2.1.1 FMT_MTD.1.1/CoreData

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

#### 5.5.2.2 FMT_MTD.1/CryptoKeys Management of TSF data

| FMT_MTD.1/CryptoKeys | Management of TSF data |
|---|---|

##### 5.5.2.2.1 FMT_MTD.1.1/CryptoKeys

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

### 5.5.3 Specification of Management Functions (FMT_SMF)

#### 5.5.3.1 FMT_SMF.1 Specification of Management Functions

| FMT_SMF.1 | Specification of Management Functions |
|---|---|

##### 5.5.3.1.1 FMT_SMF.1.1

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*

- *Ability to update the TOE, and to verify the updates using [digital signature] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- *[*
    - *Ability to start and stop services;*
    - *Ability to configure audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);*
    - *Ability to modify the behaviour of the transmission of audit data to an external IT entity;*
    - *Ability to manage the cryptographic keys;*
    - *Ability to configure the cryptographic functionality;*
    - *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;*

  *].*

## 5.5.4 Security management roles (FMT_SMR)

### *5.5.4.1 FMT_SMR.2 Restrictions on security roles*

| | |
|---|---|
| **FMT_SMR.2** | **Restrictions on Security Roles** |

#### *5.5.4.1.1 FMT_SMR.2.1*

The TSF shall maintain the roles:

- *Security Administrator.*

#### *5.5.4.1.2 FMT_SMR.2.2*

The TSF shall be able to associate users with roles.

#### *5.5.4.1.3 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.

# 5.6 Protection of the TSF (FPT)

## 5.6.1 Protection of TSF Data (Extended – FPT_SKP_EXT)

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

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

#### *5.6.1.1.1 FPT_SKP_EXT.1.1*

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

## 5.6.2 Protection of Administrator Passwords (Extended – FPT_APW_EXT)

### *5.6.2.1 FPT_APW_EXT.1 Protection of Administrator Passwords*

| FPT_APW_EXT.1 | Protection of Administrator Passwords |
|---|---|

##### *5.6.2.1.1 FPT_APW_EXT.1.1*

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

##### *5.6.2.1.2 FPT_APW_EXT.1.2*

The TSF shall prevent the reading of plaintext administrative passwords.

### 5.6.3 TSF testing (Extended – FPT_TST_EXT)

#### *5.6.3.1 FPT_TST_EXT.1 TSF Testing (Extended)*

| FPT_TST_EXT.1 | TSF Testing |
|---|---|

##### *5.6.3.1.1 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: [

*Software integrity test*<br>
*Bouncy Castle crypto module Known Answer Test*<br>
*OpenSSL crypto module Known Answer Test*

].

### 5.6.4 Trusted Update (FPT_TUD_EXT)

#### *5.6.4.1 FPT_TUD_EXT.1 Trusted Update*

| FPT_TUD_EXT.1 | Trusted Update |
|---|---|

##### *5.6.4.1.1 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 version*].

##### *5.6.4.1.2 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*].

##### *5.6.4.1.3 FPT_TUD_EXT.1.3*

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

### 5.6.5 Time stamps (Extended – FPT_STM_EXT))

#### *5.6.5.1 FPT_STM_EXT.1 Reliable Time Stamps*

| FPT_STM_EXT.1 | Reliable Time Stamps |
|---|---|

##### *5.6.5.1.1 FPT_STM_EXT.1.1*

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

##### *5.6.5.1.2 FPT_STM_EXT.1.2*

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

## 5.7 TOE Access (FTA)

### 5.7.1 TSF-initiated Session Locking (Extended – FTA_SSL_EXT)

#### *5.7.1.1 FTA_SSL_EXT.1 TSF-initiated Session Locking*

| | |
|---|---|
| **FTA_SSL_EXT.1** | **TSF-initiated Session Locking** |

##### *5.7.1.1.1 FTA_SSL_EXT.1.1*

The TSF shall, for local interactive sessions, [

- *terminate the session*]

after a Security Administrator-specified time period of inactivity.

### 5.7.2 Session locking and termination (FTA_SSL)

#### *5.7.2.1 FTA_SSL.3 TSF-initiated Termination (Refinement)*

| | |
|---|---|
| **FTA_SSL.3** | **TSF-initiated Termination** |

##### *5.7.2.1.1 FTA_SSL.3.1:*

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

#### *5.7.2.2 FTA_SSL.4 User-initiated Termination (Refinement)*

| | |
|---|---|
| **FTA_SSL.4** | **User-initiated Termination** |

##### *5.7.2.2.1 FTA_SSL.4.1:*

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

### 5.7.3 TOE Access Banners (FTA_TAB)

#### *5.7.3.1 FTA_TAB.1 Default TOE Access Banners (Refinement)*

| | |
|---|---|
| **FTA_TAB.1** | **Default TOE Access Banners** |

##### *5.7.3.1.1 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.

## 5.8 Trusted Path/Channels (FTP)

### 5.8.1 Trusted Channel (FTP_ITC)

#### *5.8.1.1 FTP_ITC.1 Inter-TSF Trusted Channel (Refinement)*

| | |
|---|---|
| **FTP_ITC.1** | **Inter-TSF Trusted Channel** |

#### *5.8.1.1.1 FTP_ITC.1.1*

The TSF shall **be capable of using [*TLS*] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [*[REST API]*, *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**.

#### *5.8.1.1.2 FTP_ITC.1.2*

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

#### *5.8.1.1.3 FTP_ITC.1.3*

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

## 5.8.2 Trusted Path (FTP_TRP)

### *5.8.2.1 FTP_TRP.1/Admin Trusted Path (Refinement)*

| | |
|---|---|
| **FTP_TRP.1/Admin** | **Trusted Path** |

#### *5.8.2.1.1 FTP_TRP.1.1/Admin*

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

#### *5.8.2.1.2 FTP_TRP.1.2/Admin*

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

#### *5.8.2.1.3 FTP_TRP.1.3/Admin*

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

# 6 Security Assurance Requirements

The TOE security assurance requirements as outlined in NDcPP v2.2e along with the refinements documented in NDcPP v.2.2e Section 7 are listed in the following table.

| 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) |

**Table 8 - TOE Security Assurance**

# 7 TOE Summary Specification

The following TSS rationales identify and describe how the Security Functional Requirements identified in section "5 Security Functional Requirements" are met by the TOE.

## 7.1 Security Audit (FAU) TSS Rationale

| Security Audit (FAU) SFR | Specification/Rationale |
|---|---|
| FAU_GEN.1 Audit data generation | The TOE generates a comprehensive set of audit logs that identify specific TOE operations whenever an auditable event occurs. Auditable events are specified at **Table 7 - FAU_GEN Auditable Events.**<br>Each of the events specified in the audit record contains required details, including identification of the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event. Cryptographic keys are identified with a reference ID.<br>The audit trail consists of the individual log entries, one audit record for each event that occurred. The TOE will drop new audit data when the local storage space for audit data is full. Administrators are instructed to monitor the log buffer to view the audit records. There are two log files which store local events. The MainLog is a local file that is limited to a total of 1,391.44 MB. The JavaLog is a local file that is limited to a total of 12,724.00 MB. The limit is specified as a file size, not a specific number of events.<br>The TOE does not have an interface to modify audit records.<br><br>Sample Audit Record (successful login):<br><182>1 2019-01-29T22:35:15.979Z nsxmanager-ob-11990881-1-test NSX 2392 SYSTEM [nsx@6876 audit="true" comp="nsx-manager" subcomp="http"] UserName="admin@10.2.103.175", ModuleName="ACCESS_CONTROL", Operation="LOGIN", Operation status="success" |
| FAU_GEN.2 User identity association | The TOE ensures that each auditable event is associated with the user that triggered the event and as a result they are traceable to a specific user. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented. Refer to the Guidance documentation for configuration syntax and information. |
| FAU_STG_EXT.1 Protected Audit Event Storage | The TOE stores its own syslog events locally on the platform, and can offload events to an audit server protected by TLS. The TOE will stop logging new audit data if the audit local space is full. The TOE does not provide a method of clearing the locally stored records.<br>When connectivity with the audit server is interrupted, the TOE will continue to store syslog events locally. The TOE will transmit any locally stored contents when connectivity to the syslog server is restored. Otherwise, TOE transmits audit events to syslog server in real-time. |

**Table 9 – TSS Rationale for Security Audit (FAU)**

## 7.2 Cryptographic Support (FCS) TSS Rationale

| Cryptographic Support (FCS) SFR | Specification/Rationale |
|---|---|
| FCS_CKM.1 Cryptographic Key Generation | Asymmetric cryptographic keys are generated also in accordance with the RSA schemes using cryptographic key sizes of 2048 bits or greater and ECC schemes using |

| | |
|---|---|
| | 'NIST curves' [P-256, P-384, P-521] that meet the FIPS 186-4, Digital Signature Standard. (CAVP Cert# A1292, C2174) |
| FCS_CKM.2 Cryptographic Key Establishment | The cryptographic key establishment is implemented in the TOE according to the RSAES-PKCS1-v1_5 based schemes for TLS, and Elliptic curve-based schemes (CAVP Cert # A1292, C2174) that meet NIST SP 800-56Ar3 for TLS. |
| FCS_CKM.4 Cryptographic Key Destruction | The TOE is designed to destroy Critical Security Parameters (CSPs) when no longer required for use to mitigate the possibility of disclosure. Volatile memory (RAM) is cleared with overwriting zeros. Non-volatile memory (virtual disk) is cleared with overwrite consisting of zeros. |
| FCS_COP.1 Cryptographic Operation / DataEncryption | The TOE supports AES encryption and decryption in CBC and GCM modes with 128-bit, and 256-bit key sizes validated as conforming to ISO 18033-3 and CBC as specified in ISO 10116 and GCM as specified in ISO 19772 (CAVP Cert# A1292, C2174). |
| FCS_COP.1 Cryptographic Operation / SigGen | The TOE supports RSA signature generation and verification according to FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3 with 2048-bit and 3072-bit key sizes utilizing SHA-1 (protocol only), SHA-256, SHA-384 (CAVP Cert# A1292, C2174). |
| FCS_COP.1 Cryptographic Operation / Hash | The TOE supports hashing using SHA-1, SHA-256, SHA-384 validated as conforming to ISO/IEC 10118-3:2004 (CAVP Cert# A1292, C2174). |
| FCS_COP.1 Cryptographic Operation / KeyedHash | The TOE supports keyed hash HMAC-SHA1, HMAC-SHA256, and HMAC-SHA-384 validated as conforming to ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2. Supported cryptographic key sizes: 160, 256, 384 bits and message digest sizes: 160, 256, 384. Keyed hash use matches the validated hash algorithms implemented by the cryptographic module (CAVP Cert# A1292, C2174). |
| FCS_RBG_EXT.1 Random Bit Generation | The TOE uses two different software-based pseudo-random number generators which are both initialized with input from multiple independent entropy sources.<br>The OpenSSL module utilizes the CTR_DRBG (AES) deterministic random bit generator that conforms to ISO/IEC 18031:2011 (CAVP Cert #A1292). The VMware Java JCE (Java Cryptographic Extension) Module uses HASH_DRBG-512 deterministic random bit generator that conforms to ISO/IEC (CAVP Cert #C2174).<br>Both are seeded with full entropy combined from a software-based noise source (the Linux Kernel Random Number Generator (LKRNG) operating in a blocking mode based on the timing variations of internal kernel-level processes), and a hardware-based noise source (Intel RDSEED and RDRAND instructions). All entropy is extracted, processed, and accumulated by LKRNG from the mixed noise sources. Accumulated entropy is not preserved across system reboots. |
| FCS_TLSC_EXT.1 TLS Client Protocol Without Mutual Authentication | The TOE as a client allows trusted channel using TLS 1.1, conformant to RFC 4346 and TLS 1.2, conformant to RFC 5246. The TOE is restricted to the following ciphersuites:<br>TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289<br>TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289<br>TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492<br>TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492<br>TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289<br>TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289<br>TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288<br>TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246<br>TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268<br>TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288<br>TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246<br>TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268<br><br>The following reference identifiers are supported: |

| | |
|---|---|
| | • CN<br>• SAN (not mandated)<br>• FQDN<br>Wildcards are also supported.<br><br>The TLS channel is terminated if verification fails.<br><br>Also, TSF shall deny connections with server requesting SSL 2.0, SSL3.0, and TLS1.0 programmatically.<br><br>The TLS client will transmit the Supported Elliptic Curves extension in the Client Hello message by default with support for the following NIST curves: secp256r1, secp384r1, and secp521r1 (CAVP Cert #A1292, C2174). The non-TOE server can choose to negotiate the elliptic curve from this set for any of the mutually negotiable elliptic curve ciphersuites and no additional configuration is required. The TOE also supports key agreement using the server's RSA public key 2048 bits or 3072 bits (CAVP Cert #A1292, C2174). |
| FCS_TLSC_EXT.2 TLS Client Support for Mutual Authentication | The TOE Audit Server trusted channel implements TLS 1.1, conformant to RFC 4346 and TLS 1.2, conformant to RFC 5246. The TOE supports mutual X509v3 certificate-based authentication.<br>The following reference identifiers are supported:<br>• CN<br>• SAN<br>• FQDN<br>Wildcards are also supported. |
| FCS_TLSS_EXT.1 TLS Server Protocol | The TOE REST API implements TLS 1.1, conformant to RFC 4346 and TLS 1.2, conformant to RFC 5246. The TOE supports the following ciphersuites:<br>TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289<br>TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289<br>TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492<br>TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492<br>TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289<br>TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289<br>TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288<br>TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246<br>TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268<br>TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288<br>TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246<br>TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268<br><br>Also, TSF shall deny connections from clients requesting SSL 2.0, SSL3.0, and TLS1.0 programmatically. |
| | The TOE establishes session keys for TLS sessions using RSA [2048, 3072 bits] ECDHE curves [secp256r1, secp384r1, secp521r1] and no other curves]] (CAVP Cert #A1292, C2174). The TOE does not support configuration of key/curve used during key agreement.<br>The TOE does not support session resumption or session tickets. |

**Table 10 – TSS Rationale for Cryptographic Support (FCS)**

## 7.3 Identification and Authentication (FIA) TSS Rationale

| Identification and Authentication (FIA) SFR | Specification/Rationale |
|---|---|

| | |
|---|---|
| FIA_AFL.1 Authentication Failure Management | A user account becomes locked after a security administrator-configurable (1 to 5) number of unsuccessful authentication attempts. Once the user account is locked out, all further authentication attempts are reported as unsuccessful, even when correct information is provided. To regain access, the user has to wait an administrator-configurable time duration before being allowed to successfully authenticate.<br>The TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, by distinguishing between local and remote login attempts through a whitelist of “local” IP addresses using a dedicated port. |
| FIA_PMG_EXT.1 Password Management | The passwords can be composed of any combination of upper and lower case letters, numbers, and the following special characters: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”. The minimum password length is settable by the Security Administrator, and can be configured for minimum password lengths between 8 and 20 characters. |
| FIA_UIA_EXT.1 User Identification and Authentication<br>FIA_UAU_EXT.2 Password-based Authentication Mechanism | The TOE requires all users to be successfully identified and authenticated before allowing any REST API commands (other than the display of the warning banner). A pre-authentication banner is also displayed by the UI and CLI at the login prompt. The UI utilizes the REST API for all TOE functionalities.<br>The REST API uses an HTTPS session, requiring user name and password be provided and successfully verified prior to access being granted. The reference password value is a salted-iterated hash that is stored locally. The supplied user credentials are similarly processed and success requires a match with the reference password value. The same process applies for the local CLI. Connections are treated as remote, unless the IP address in in a whitelist of “local” addresses.<br>The local administration is terminated by exiting the shell by typing the command ‘exit’. |
| FIA_UAU.7 Protected Authentication Feedback | When a user enters their password using the UI, only ‘*’ characters are displayed. The user credentials are protected by a secure channel for the REST API. |
| FIA_X509_EXT.1 X.509 Certificate Validation | The TOE supports the use of X.509v3 certificates as defined by RFC 5280 to mutually authenticate external IT entities. When a X509 certificate is presented during a TLS handshake, the TOE verifies the trust chain, performing validation of the certificates and carries out revocation checking of each certificate. The revocation check is performed by checking against a valid CRL. If the TOE does not contain a valid CRL and cannot retrieve a valid CRL from one of the CDPs, the session will not be established. |
| FIA_X509_EXT.2 X509 Certificate Authentication | The TOE supports the use of X.509v3 certificates as defined by RFC 5280 to authenticate connections with authorized IT entities. When certificate based authentication is used for any of the trusted channels, the TOE validates the presented certificate, checking its chain of trust against the TOE’s internal trusted store, and performs a certificate revocation check. Certificate validation includes path validation (including checking CA certificates) certificate processing (including validating the extendedKeyUsage field), and extension processing (including checking the BasicConstraints extension). Verifying the chain of trust includes validating each certificate in the chain, verifying that certificate path consists of trusted CA certificates, and performing revocation checks on all certificates in the path. Revocation checking is implemented using CRL. If any part of the authentication fails, the connection is terminated at the handshake stage. The TOE does not provide user configurable options for X.509v3 validation.<br>The TOE performs certificate validation when establishing TLS connections for Audit Server.<br>The TOE, through the TLS protected REST API, accepts certificates based on an internally generated CSR, or a private key and corresponding certificate. The certificates are validated before use, checking:<br>• The Basic constrains extension with the CA flag is appropriately set<br>• The Key usage extension with the “keyCertSign” bit is set |

| | |
|---|---|
| | - The Certificate is not expired.<br>All certificates are stored in a private, persistent location on the TOE. |
| FIA_X509_EXT.3 X.509 Certificate Requests | The product supports multiple types of certificates. The TOE relevant certificate types do not contain “device specific information”. The certificate supports Common Name, Organization, Organizational Unit, and Country |

**Table 11 – TSS Rationale for Identification and Authentication (FIA)**

## 7.4 Security Management (FMT) TSS Rationale

| **Security Management (FMT) SFR** | **Specification/Rationale** |
|---|---|
| FMT_MOF.1/ManualUpdate | The TOE restricts the ability to enable the functions to perform manual update to the Security Administrator. |
| FMT_MOF.1/Services | The TOE provides all the capabilities necessary to securely manage the TOE, and the services provided by the TOE. The specific management capabilities available from the TOE are identified in the text of the FMT_SMF.1. The Security administrator have the ability to generate, delete and import/export cryptographic keys. The management functionality of the TOE is provided through the remote administration interface via REST API.<br>TOE supports both local and remote administration. The TOE provides a local console interface which supports CLI. REST API over TLS is the remote interface. |
| FMT_MOF.1.1/Functions | Only authenticated System Administrator are allowed to modify the behavior of syslog server or get log data from the system. |
| FMT_MTD.1.1/CoreData | Only authenticated System Administrator can manage the TSF data and thus it is protected.<br>There are two types of administrative users within the system:<br>Security Administrator - Can perform four types of operations, namely, create, read, update, and delete<br>Auditor - can only perform read operation. |
| FMT_MTD.1.1/CryptoKeys | Only the authenticated System Administrators can manage the cryptographic keys in the TOE. |
| FMT_SMF.1 | The TSF shall be capable of performing the following management functions:<br>• *Ability to administer the TOE locally and remotely;*<br>• *Ability to configure the access banner;*<br>• *Ability to configure the session inactivity time before session termination or locking;*<br>• *Ability to update the TOE, and to verify the updates using [digital signature] capability prior to installing those updates;*<br>• *Ability to configure the authentication failure parameters for FIA_AFL.1;*<br>• *[*<br>○ *Ability to start and stop services;*<br>○ *Ability to configure audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);*<br>○ *Ability to modify the behaviour of the transmission of audit data to an external IT entity;*<br>○ *Ability to manage the cryptographic keys;*<br>○ *Ability to configure the cryptographic functionality;*<br>○ *Ability to set the time which is used for time-stamps;*<br>○ *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;*<br>*].*<br>All these functions are performed by the Administrator via the REST API interface. |
| FMT_SMR.2 Restrictions on security roles | The TOE supports the Security Administrator role. The Security Administrator can change their own password. |

**Table 12 – TSS Rationale for Security Management (FMT)**

## 7.5 Protection of the TSF (FPT) TSS Rationale

| **Protection of the TSF (FPT) SFR** | **Specification/Rationale** |
|---|---|
| FPT_SKP_EXT.1 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) | The TOE protects Critical Security Parameters (CSP) such as cryptographic keys so they are not directly accessible via normal administrative interfaces. |
| FPT_APW_EXT.1 Protection of Administrator Passwords | The TOE protects Critical Security Parameters (CSP) such as stored passwords so they are not directly accessible via normal administrative interfaces. The passwords are protected using a salted, iterated SHA-256 hash (Linux PAM), the resulting reference password values are stored locally in a limited access file. |
| FPT_TST_EXT.1 TSF Testing | The TOE performs diagnostic self-tests during start-up and generates audit records to capture any failures. These self-tests comply with the FIPS 140-2 requirements for self-testing. The modules perform known-answer algorithm testing, and integrity testing. These self-tests cover all anticipated modes of failure, and therefore are sufficient such that the TSF operates correctly. Failure of any of the FIPS mode tests during the boot process will stop the start-up process and prompt the user to reload. For all start-up tests, successful completion is indicated by the TOE reaching operational status.<br><br>Start-up tests:<br>• Software integrity test<br>• Bouncy Castle crypto module Known Answer Test<br>• OpenSSL crypto module Known Answer Test |
| FPT_TUD_EXT.1 Trusted Update | The current version of the TOE can be obtained through the REST API. When updates become available, a Security Administrator can manually download the update from VMware.com and initiate the update process. The TOE image files are digitally signed using a RSA mechanism so their integrity can be verified prior to the update process, and an image that fails an integrity check will not be loaded. The digital certificates used by the update verification mechanism are contained on the TOE. |
| FPT_STM_EXT.1 Reliable Time Stamps | The TOE provides a source of date and time information, used in audit timestamps. The clock function is reliant on the system clock provided by the underlying hardware. The following TOE functionality uses the system clock for dependable date and time information:<br>• checking expiration of X509 certificate<br>• audit events<br>• user authentication session expiration<br>This function can be configured through the REST API by a Security Administrator. |

**Table 13 – TSS Rationale for Protection of the TSF (FPT)**

## 7.6 TOE Access (FTA) TSS Rationale

| **TOE Access (FTA) SFR** | **Specification/Rationale** |
|---|---|

| | |
|---|---|
| FTA_SSL_EXT.1 TSF-initiated Session Locking<br>FTA_SSL.3 TSF-initiated Termination | A Security Administrator uses the REST API or CLI to configure maximum inactivity times for administrative sessions. When a session is inactive (i.e., no session input) for the configured period of time) the TOE terminates the session and requires the administrator to log in again when a new session is needed.<br>The administrative session terminates the local console or remote-client TLS connection when there is no activity for the 'configured period of time' as described above. The TOE will terminate the session in both cases. |
| FTA_SSL.4 User-initiated Termination | A Security Administer logouts of the REST API session by calling a method that requests the server to destroy the session. The UI provides a logout interface which calls the same method. For CLI sessions, the user may terminate a session by typing 'exit'. |
| FTA_TAB.1 Default TOE Access Banners | The REST API provides access to a configurable banner without requiring user authentication. The UI displays the configurable banner before user login. |

**Table 14 – TSS Rationale for TOE Access (FTA)**

## 7.7 Trusted Path/Channel (FTP) TSS Rationale

| Trusted Path/Channel (FTP) SFR | Specification/Rationale |
|---|---|
| FTP_ITC.1 Inter-TSF trusted channel | The TOE protects communications with the following devices using the TLS v1.1 or v1.2 protocol.<br>External audit server – TOE is a TLS client that validates the server certificate chain. |
| FTP_TRP.1/Admin Trusted Path | All remote administrative communications take place through a TLS/HTTPS session, protected using AES encryption. The Security Administrator can authenticate the TOE by validating the RSA host key pair generated by the TOE or generated elsewhere and imported into the TOE. |

**Table 15 – TSS Rationale for Trusted Path/Channel (FTP)**

## 7.8 Key Storage and Zeroization

The following table describes the origin, storage and zeroization of keys as relevant to FCS_CKM.4 and FPT_SKP_EXT.1 provided by the TOE.

| Key | Key Size | Generation/Input | Output | Storage | Zeroization | Use |
|---|---|---|---|---|---|---|
| RSA Private Key | 2048 - 3072 bits | Internally generated using OpenSSL, or input into system though REST API | Associated cert can be output, but not the private key itself. | File System, keystore | Overwritten by zeros before the disk block is re-allocated | Signature generation, decryption<br>Used by calling application |
| RSA Public Key | 2048 - 3072 bits | Internally generated using OpenSSL, or input into system through REST API | Output via REST API in plaintext (in the form of cert) | File System, keystore | Overwritten by zeros before the disk block is re-allocated | Signature verification, encryption<br>Used by calling application |
| AES Key | 128, 256 bits | Internally generated using API | Never | In RAM | Overwritten by zeros when deleted; or cleared when power cycled or host reboots | Encryption, Decryption |

| Key | Key Size | Generation/Input | Output | Storage | Zeroization | Use |
|---|---|---|---|---|---|---|
| AES GCM Key | 128, 256 bits | Internally generated using API | Never | In RAM | Overwritten by zeros when deleted; or cleared when power cycled or host reboots | Encryption, Decryption, MAC Generation and Verification |
| HMAC key | 160, 224, 256, 384 bits | Internally generated using API | Never | In RAM | Overwritten by zeros when deleted; or cleared when power cycled or host reboots | Message Authentication |
| CTR_DRBG CSPs | V (128 bits); Key (128, 190, 256 bits); Entropy Input | Internally generated using API | Never | In RAM | Overwritten by zeros when deleted; or cleared when power cycled or host reboots | Random Number Generation |

**Table 16 - Key Storage and Zeroization**

# 8 Glossary

| | |
|---|---|
| AES | Advanced Encryption Standard |
| ANSI | American National Standards Institute |
| API | Application Program Interface |
| cPP | collaborative Protection Profile |
| CCM | Counter with Cipher Block Chaining-Message Authentication Code |
| CM | Configuration Management |
| CSP | Critical security parameter |
| DH | Diffie Hellman |
| EAL | Evaluation Assurance Level |
| ECC | Elliptic Curve Cryptography |
| ECDSA | Elliptic Curve Digital Signature Algorithm |
| FIPS | Federal Information Processing Standard |
| HMAC | Keyed-Hash Authentication Code |
| I&A | Identification and Authentication |
| ID | Identification |
| IP | Internet Protocol |
| IPv6 | Internet Protocol Version 6 |
| ISO | International Organization for Standardization |
| IT | Information Technology |
| NDcPP | Network Device collaborative Protection Profile |
| PP | Protection Profile |
| RFC | Request for Comment |
| RNG | Random Number Generator |
| RSA | Rivest, Shamir, Adelman |
| SFR | Security Functional Requirement |
| SHA | Secure Hash Algorithm |
| ST | Security Target |
| TCP/IP | Transmissions Control Protocol/ Internet Protocol |
| TOE | Target of Evaluation |
| TSF | TSF interfaces |
| TSFI | TSF interfaces |
| UDP | User Datagram Protocol |
| VM | Virtual Machine |