# Citrix ADC (MPX FIPS and VPX FIPS) Version 12.1 Security Target

Document Version: 1.6

intertek
acumen
security

2400 Research Blvd
Suite 395
Rockville, MD 20850

## Revision History:

| Version | Date | Changes |
|---|---|---|
| Version 1.0 | 4/15/2021 | Finalized version for submission to NIAP |
| Version 1.1 | 5/27/2021 | Updated TDs. |
| Version 1.2 | 12/8/2021 | Addressing validator comments and finalization |
| Version 1.3 | 1/12/2022 | Updated based on validator feedback |
| Version 1.4 | 1/19/2022 | Updated based on validator feedback |
| Version 1.5 | 1/24/2022 | Updated to update AGD reference for new version |
| Version 1.6 | 1/24/2022 | Updated to update AGD reference for new version |

# Contents

# 1 Introduction

The Security Target (ST) serves as the basis for the Common Criteria (CC) evaluation and identifies the Target of Evaluation (TOE), the scope of the evaluation, and the assumptions made throughout. This document will also describe the intended operational environment of the TOE, and the functional and assurance requirements that the TOE meets.

## 1.1 Security Target and TOE Reference

This section provides the information needed to identify and control the TOE and the ST.

**Table 1 – TOE/ST Identification**

| Category | Identifier |
|---|---|
| ST Title | Citrix ADC (MPX FIPS and VPX FIPS) Version 12.1 Security Target |
| ST Version | 1.6 |
| ST Date | 1/24/2022 |
| ST Author | Acumen Security, LLC |
| TOE Identifier | Citrix ADC (MPX FIPS and VPX FIPS) Version 12.1 |
| TOE Version | 12.1 |
| TOE Developer | Citrix Systems Inc. |
| Key Words | Network Device, Security Appliance |

## 1.2 TOE Overview

The Citrix Application Delivery Controller (ADC) are purpose-built networking appliances whose function is to improve the performance, security and resiliency of applications delivered over the web. The ADC intelligently distributes, optimizes application performance, enhances application availability with advanced Layer 4 – Layer 7 load balancing, secures applications from attacks, and lowers server expenses by offloading computationally intensive tasks. The TOE comprises Citrix ADC 12.1 software running on the following:

- Physical Platforms
  - MPX 8900 FIPS
  - MPX 15000-50G FIPS
- Virtual Platforms
  - VPX FIPS on ESXi 6.5 running on a Dell PowerEdge R630 Server

Citrix ADC MPX FIPS & Citrix ADC VPX FIPS are network devices and virtual network devices that combine Layer 4 - Layer 7 load balancing and content switching with application acceleration, data compression, static and dynamic content caching, SSL acceleration, network optimization, application performance monitoring, application visibility, and robust application security via an application firewall. The Citrix ADC MPX FIPS & Citrix ADC VPX FIPS appliances support all the NIST-approved FIPS 140-2 algorithms. The evaluation is limited to the security functionality defined in the SFRs.

## 1.3 TOE Description

### 1.3.1 TOE Evaluated Configuration

The TOE evaluated configuration consists of the physical platforms, MPX 8900 FIPS and MPX 15000-50G FIPS. Both, the MPX 8900 FIPS and the MPX 15000-50G FIPS, operate using the Intel® Xeon E5-2620 v4 (Broadwell) processor. Additionally, the evaluated configuration includes the VPX FIPS virtual platform.

This virtual platform is hosted within a Dell PowerEdge R630 Server running an instance of VMware ESXi 6.5 hypervisor. The VPX is hosted on a server which operates on an Intel® Xeon E5-2680 v4 (Broadwell) processor. FreeBSD 8.4 is the operating system on all the physical and virtual platforms.

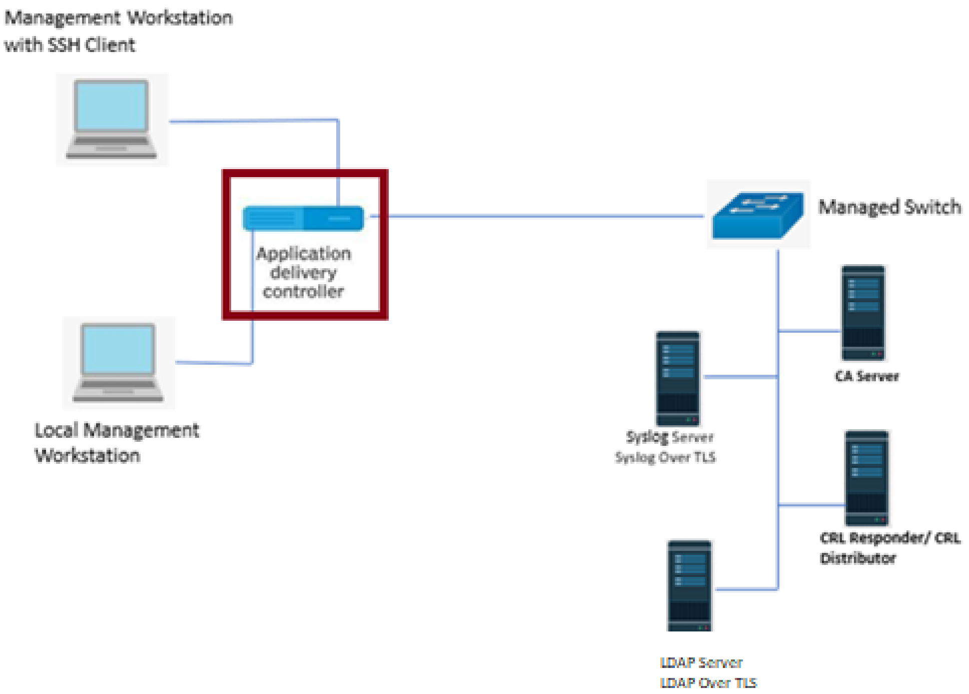

**Figure 1 – Representative TOE Deployment**

The above figure represents a typical deployment of the TOE. The TOE also supports (sometimes optionally) secure connectivity with several other IT environment devices as described in Section 1.4.

### 1.3.2 Physical Boundaries

The TOE is a hardware and software solution that is comprised of the security appliance models described above in Section 1.3. The TOE is represented by the red box in the above figure. For VPX, as defined in the NDcPP 2.2e case 1 applies.

### 1.3.3 Logical Boundaries

The TOE provides the security functions required by NDcPP v2.2e. The TOE is composed of the Citrix ADC OS running directly on the MPX appliance hardware and VPX Running on ESXi 6.5. It is also comprised of several security features. Each of the security features identified above consists of several security functionalities, as identified below.

- Security Audit
- Cryptographic Support
- Identification and Authentication
- Security Management
- Protection of the TSF
- TOE Access
- Trusted path/channels

These security functions are discussed in detail in the sections below.

### 1.3.3.1 Security Audit

The TOE keeps local and remote audit records of security relevant events. Remote audit records are transferred via TLS to the external audit server.

### 1.3.3.2 Cryptographic Support

The TOE provides cryptographic support for the SSH for remote administrative access and TLS connections to external IT devices.

**Table 2 – CAVP Algorithm Certificate References**

| Algorithm | Description/ Operation | Supported Mode/ Standard | CAVP Cert. # | SFR |
|---|---|---|---|---|
| RSA | Signature Generation, Verification, and key transport | FIPS PUB 186-4 | C1918<br>C1920 | FCS_CKM.1<br>FCS_CKM.2<br>FCS_COP.1/SigGen<br>FCS_COP.1/SigVer |
| ECDSA | EC Signature services in support of SSH and TLS authentication | FIPS PUB 186-4 | C1918<br>C1920 | FCS_CKM.1<br>FCS_COP.1/SigGen<br>FCS_COP.1/SigVer |
| AES | Encryption in support of TLS and SSH protocols | ISO 18033-3<br>ISO 10116<br>ISO 19772 | C1918<br>C1920 | FCS_COP.1/ DataEncryption |
| SHA | Cryptographic hashing services | ISO/IEC 10118-3:2004 | C1918<br>C1920 | FCS_COP.1//Hash |
| HMAC | Keyed hashing services | ISO/IEC 9797-2:2011 | C1918<br>C1920 | FCS_COP.1/KeyedHash |
| DRBG | Random number generation | ISO/IEC 18031:2011 | C1918<br>C1920 | FCS_RBG_EXT.1 |
| KAS ECC | Key Agreement | NIST Special Publication 800-56A Revision 3 | A1919<br>A1920 | FCS_CKM.2 |

The cryptography for the TOE is provided by Citrix ADC CP Cryptographic Library v3.0 and Citrix ADC CP Cryptographic Library v4.0 running on FreeBSD 8.4. This is the underlying OS of the TOE on which the firmware runs.

### 1.3.3.3 Identification and Authentication

The TOE provides two types of authentication to provide a trusted means for Security Administrators and remote endpoints to interact:

- Password-based or public-key authentication for Security Administrators
- X.509v3 certificate-based authentication for remote devices

Device-level authentication allows the TOE to establish a secure communication channel with a remote endpoint. Security Administrators can set a minimum length for passwords (between 4 and 127 characters). Additionally, the TOE detects and tracks consecutive unsuccessful remote authentication attempts and will prevent the offending attempts from authenticating when a Security Administrator defined threshold is reached.

#### 1.3.3.4 Security Management

The TOE enables secure local and remote management of its security functions, including:

- Local console CLI administration
- Remote CLI administration via SSHv2
- Administrator authentication using a local database
- Timed user lockout after multiple failed authentication attempts
- Password complexity enforcement
- Role Based Access Control - the TOE supports several types of administrative user roles. Collectively these sub-roles comprise the "Security Administrator"
- Configurable banners to be displayed at login
- Timeouts to terminate administrative sessions after a set period of inactivity
- Protection of secret keys and passwords

#### 1.3.3.5 Protection of the TSF

The TOE ensures the authenticity and integrity of software updates through hash comparison and requires administrative intervention prior to the software updates being installed.

#### 1.3.3.6 TOE Access

Prior to login, the TOE displays a banner with a message configurable by the Security Administrator. The TOE terminates user connections after an Authorized Administrator configurable amount of inactivity time.

#### 1.3.3.7 Trusted Path/Channels

The TOE uses TLS to provide a trusted channel between itself and remote syslog and LDAP servers. The TOE uses SSH to provide a trusted path between itself and remote administrators.

### 1.3.4 TOE Documentation

The following documents are essential to understanding and controlling the TOE in the evaluated configuration:

- Citrix ADC (MPX FIPS & VPX FIPS) Security Target 1.6, January 24, 2022
- Citrix ADC (MPX FIPS & VPX FIPS) Common Criteria Guidance, 1.4, January 24, 2022.

The TOE guidance documentation can be found at:

https://docs.citrix.com/en-us/citrix-adc/12-1/

- MPX FIPS: https://docs.citrix.com/en-us/citrix-adc/12-1/ssl/citrix-adc-mpx-fips-certified-appliance.html

- VPX FIPS: https://docs.citrix.com/en-us/citrix-adc/12-1/ssl/citrix-adc-vpx-fips-appliances.html

## 1.4 TOE Environment

The following environmental components are required to operate the TOE in the evaluated configuration:

**Table 3 – Required Environmental Components**

| Components | Description |
|---|---|
| Local Management Workstation | This include any IT Environment Management workstation that is directly connected to the TOE. |
| Management Workstation with SSH Client | This includes any IT Environment Management workstation with an SSH client installed that is used by the TOE administrator to support TOE administration through SSH protected channels. Any SSH client that supports SSHv2 may be used. |
| Syslog server | The syslog audit server is used for remote storage of audit records that have been generated by and transmitted from the TOE. The syslog server must support communications using TLS 1.1 or TLS 1.2. |
| Certificate Authority | This is the remote CA used for various certificate related operations, such as, importing certificates. |
| LDAP Server | The LDAP server is used for authentication of administrator credentials. The LDAP server must support communications using TLS 1.1 or TLS 1.2. |
| CRL Responder/CRL Distributer | Server which contains updated revocation list for the TOE. |
| Managed Switch | NOTE: While this is not required for proper TOE operation. Switches are often found in the IT environment (and is included in figure 1 above). |

## 1.5 Product Functionality not Included in the Scope of the Evaluation

Hardware and software located in the TOE environment (see Section 1.4) are not included in the scope of evaluations against this Security Target.

Only security functionality specified in the SFRs in Section 5.2 (and the corresponding security functions in Section 6) is covered by the scope of evaluation against this Security Target. Other product features or functionality are considered unevaluated, because they are not included in the scope of this Security Target:

- Web Logging
- Application Firewall
- Global Server Load Balancing (GSLB)
- AAA-TM Authentication
- External authentication methods: Kerberos, TACACS+, SAML, RADIUS
- Responder
- Rewrite (URL Transformation)
- Layer 3 Routing
- Vpath
- RISE
- High Availability
- Cloud Bridge
- CallHome

- Integrated Disk Caching
- General TLS VPN functionality
- Clientless VPN functionality
- SSL acceleration – SSL termination for application servers
- AppFlow
- AppQoE
- BGP
- Cache Redirection
- Compression Control
- Content Accelerator
- Content Filtering
- Content Switching
- FEO
- OSPF
- LSN
- RDP Proxy
- RIP
- HTM Injection
- Http DoS Protection
- Integrated Caching
- Surge Protection
- ISIS
- Priority Queuing
- Reputation
- Sure Connect
- NetScaler Push
- ContentInspection
- Connection Quality Analytics
- Adaptive TCP
- Forward Proxy
- Video Optimization
- URL Filtering

Additionally, the following features may not be used when the TOE is operated in a manner compliant with this Security Target:

- IPv6
- NTP based updates to the time
- Use of superuser privileges except as described in [CCECG]
- ADC GUI (HTTP/HTTPS), ADC Nitro API and ADM

# 2 Conformance Claims

This section identifies the TOE conformance claims, conformance rational, and relevant Technical Decisions (TDs).

## 2.1 CC Conformance Claims

The TOE is conformant to the following:

- 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 3, Version 3.1, Revision 5, April 2017: Part 3 conformant

## 2.2 Protection Profile Conformance

This ST claims exact conformance to the Collaborative Protection Profiles for Network Devices, Version 2.2e, March 12, 2020.

## 2.3 Conformance Rationale

This Security Target provides exact conformance to NDcPP v2.2e. 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.

### 2.3.1 Technical Decisions

All NIAP Technical Decisions (TDs) issued to date and applicable to NDcPP v2.2e have been considered. Table  identifies all applicable TDs.

**Table 4 – Relevant Technical Decisions**

| Technical Decision | Applicable (Y/N) | Exclusion Rationale (if applicable) |
|---|---|---|
| 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 Elliptic curve-based key establishment and NIST SP 800-56Arev3 | Yes | |
| TD0580 - NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e | Yes | |
| TD0572 - NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers | Yes | |
| TD0571 - NiT Technical Decision for Guidance on how to handle FIA_AFL.1 | Yes | |
| TD0570 - NiT Technical Decision for Clarification about FIA_AFL.1 | Yes | |
| TD0569 - NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7 | No | The evaluation does not include FCS_TLSS_EXT.1. Therefore, this is not applicable. |
| TD0564 - NiT Technical Decision for Vulnerability Analysis Search Criteria | Yes | |
| TD0563 - NiT Technical Decision for Clarification of audit date information | Yes | |
| TD0556 - NIT Technical Decisions for RFC 5077 question | No | The evaluation does not include FCS_TLSS_EXT.1. Therefore, this is not applicable. |
| TD0555 - NIT Technical Decision for RFC Reference incorrect in TLSS Test | No | The evaluation does not include FCS_TLSS_EXT.1. Therefore, this is not applicable. |
| TD0547 - NIT Technical Decision for Clarification on developer disclosure of AVA_VAN | Yes | |
| TD0546 - NIT Technical Decision for DTLS – clarification of Application Note 63 | No | The evaluation does not include FCS_DTLSC_EXT.1. Therefore, this is not applicable. |
| TD0538 - NIT Technical Decision for Outdated link to allowed-with list | Yes | |
| TD0537 - NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3 | No | The evaluation does not include FCS_TLSC_EXT.2, therefore, this is not applicable. |
| TD0536 - NIT Technical Decision for Update Verification Inconsistency | Yes | |
| TD0528 - NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4 | No | The evaluation does not include FCS_NTP_EXT.1, |
| TD0527 - Updated to Certificate Revocation Testing (FIA_X509_EXT.1) | Yes | |

# 3 Security Problem Definition

The security problem definition has been taken directly from the claimed PP and any relevant EPs/Modules/Packages specified in Section 2.2 and is reproduced here for the convenience of the reader. The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any Organizational Security Policies (OSPs) that the TOE is expected to enforce.

## 3.1 Threats

The threats included in Table 5 are drawn directly from the PP and any EPs/Modules/Packages specified in Section 2.2.

**Table 5 – Threats**

| ID | Threat |
|---|---|
| T.UNAUTHORIZED_ADMINISTRATOR_ACCESS | Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides. |
| T.WEAK_CRYPTOGRAPHY | Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort. |
| T.UNTRUSTED_COMMUNICATION_CHANNELS | Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself. |
| T.WEAK_AUTHENTICATION_ENDPOINTS | Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g., a shared password that is guessable or transported as plaintext. The consequences are |

| ID | Threat |
|---|---|
| | the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised. |
| T.UPDATE_COMPROMISE | Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration. |
| T.UNDETECTED_ACTIVITY | Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised |
| T.SECURITY_FUNCTIONALITY_COMPROMISE | Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker. |
| T.PASSWORD_CRACKING | Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic and may allow them to take advantage of any trust relationships with other Network Devices. |
| T.SECURITY_FUNCTIONALITY_FAILURE | An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device. |

## 3.2 Assumptions

The assumptions included in Table 6 are drawn directly from PP and any relevant EPs/Modules/ Packages.

**Table 6 – Assumptions**

| ID | Assumption |
|---|---|
| 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. |
| 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).<br>If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform. |
| A.NO_THRU_TRAFFIC_PROTECTION | A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall). |

| ID | Assumption |
|---|---|
| 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.<br>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). |
| A.REGULAR_UPDATES | The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities. |
| A.ADMIN_CREDENTIALS_SECURE | The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside. |
| A.RESIDUAL_INFORMATION | The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g., cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. |
| A.VS_TRUSTED_ADMINISTRATOR | 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. |
| A.VS_REGULAR_UPDATES | 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. |
| A.VS_ISOLATION | 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. |
| A.VS_CORRECT_CONFIGURATION | 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 OSPs included in Table 7 are  drawn directly from the PP and any relevant EPs/Modules/Packages.

**Table 7 – OSPs**

| ID | OSP |
|---|---|
| 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

The security objectives have been taken directly from the claimed PP and any relevant EPs/Modules/Packages and are reproduced here for the convenience of the reader.

## 4.1 Security Objectives for the Operational Environment

Security objectives for the operational environment assist the TOE in correctly providing its security functionality. These objectives, which are found in the table below, track with the assumptions about the TOE operational environment.

**Table 8 – Security Objectives for the Operational Environment**

| ID | Objectives for the Operational Environment |
|---|---|
| OE.PHYSICAL | Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment. |
| OE.NO_GENERAL_PURPOSE | There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS. |
| OE.NO_THRU_TRAFFIC_PROTECTION | The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment. |
| OE.TRUSTED_ADMN | 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.<br>For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE’s trust store in case such certificate can no longer be trusted. |
| OE.UPDATES | The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities. |
| OE.ADMIN_CREDENTIALS_SECURE | The Administrator’s credentials (private key) used to access the TOE must be protected on any other platform on which they reside. |
| OE.RESIDUAL_INFORMATION | The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment. |

| ID | Objectives for the Operational Environment |
|---|---|
| OE.VM_CONFIGURATION | For vNDs, the Security Administrator ensures that the VS and VMs are configured to<br>• 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<br>• correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting).<br>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.<br>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 Requirements

This section identifies the Security Functional Requirements (SFRs) for the TOE. The SFRs included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revisions 5, September 2017, and all international interpretations.

**Table 9 – SFRs**

| Requirement | Description |
|---|---|
| FAU_GEN.1 | Audit Data Generation |
| FAU_GEN.2 | User Identity Association |
| FAU_STG_EXT.1 | Protected Audit Event Storage |
| FCS_CKM.1 | Cryptographic Key Generation |
| FCS_CKM.2 | Cryptographic Key Establishment |
| FCS_CKM.4 | Cryptographic Key Destruction |
| FCS_COP.1/DataEncryption | Cryptographic Operation (AES Data Encryption/Decryption) |
| FCS_COP.1/SigGen | Cryptographic Operation (Signature Generation and Verification) |
| FCS_COP.1/Hash | Cryptographic Operation (Hash Algorithm) |
| FCS_COP.1/KeyedHash | Cryptographic Operation (Keyed Hash Algorithm) |
| FCS_RBG_EXT.1 | Random Bit Generation |
| FCS_SSHS_EXT.1 | SSH Server Protocol |
| FCS_TLSC_EXT.1 | TLS Client Protocol without Mutual Authentication |
| FIA_AFL.1 | Authentication Failure Management |
| FIA_PMG_EXT.1 | Password Management |
| FIA_UIA_EXT.1 | User Identification and Authentication |
| FIA_UAU_EXT.2 | Password-based Authentication Mechanism |
| FIA_UAU.7 | Protected Authentication Feedback |
| FIA_X509_EXT.1/Rev | X.509 Certificate Validation |
| FIA_X509_EXT.2 | X.509 Certificate Authentication |
| FMT_MOF.1/ManualUpdate | Management of Security Functions Behavior |
| FMT_MTD.1/CoreData | Management of TSF Data |
| FMT_SMF.1 | Specification of Management Functions |
| FMT_SMR.2 | Restrictions on security roles |
| FPT_SKP_EXT.1 | Protection of TSF Data (for reading of all symmetric keys) |
| FPT_APW_EXT.1 | Protection of Administrator Passwords |
| FPT_TST_EXT.1 | TSF Testing |
| FPT_STM_EXT.1 | Reliable Time Stamps |
| FPT_TUD_EXT.1 | Trusted Update |
| FTA_SSL.3 | TSF-initiated Termination |
| FTA_SSL.4 | User-initiated Termination |
| FTA_SSL_EXT.1 | TSF-initiated Session Locking |
| FTA_TAB.1 | Default TOE Access Banner |
| FTP_ITC.1 | Inter-TSF Trusted Channel |
| FTP_TRP.1/Admin | Trusted Path |

## 5.1 Conventions

The CC allows the following types of operations to be performed on the functional requirements: assignments, selections, refinements, and iterations. The following font conventions are used within this document to identify operations defined by CC:

- Assignment: Indicated with *italicized* text;
- Refinement: Indicated with **bold** text;

- Selection: Indicated with *Italicized* text;
- Iteration: Indicated by appending the iteration identifier after a slash, e.g., /SigGen.
- Where operations were completed in the PP and relevant EPs/Modules/Packages, the formatting used in the PP has been retained.
- Extended SFRs are identified by the addition of "EXT" after the requirement name.

## 5.2 Security Functional Requirements

This section includes the security functional requirements for this ST.

### 5.2.1 Security Audit (FAU)

#### 5.2.1.1 FAU_GEN.1 Audit Data Generation

**FAU_GEN.1.1**

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

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

b) Auditable events for the <u>not specified</u> 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).*
- *[<u>no other actions</u>];*

d) *Specifically defined auditable events listed in Table 10.*

**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 10.*

**Table 10 – Security Functional Requirements and 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 |

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FCS_SSHS_EXT.1 | Failure to establish an SSH session | Reason for failure |
| FCS_TLSC_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 |
| FMT_MOF.1/ManualUpdate | Any attempt to initiate a manual update | None |
| FMT_MTD.1/CoreData | 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_STM_EXT.1 | Discontinuous changes to time - either Administrator actuated or changed via an automated process.<br>(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). |
| FPT_TUD_EXT.1 | Initiation of update; result of the update attempt (success or failure) | 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_SSL_EXT.1 (if “terminate the session” is selected) | The termination of a local session by the session locking mechanism | None |
| FTA_TAB.1 | None | None |

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| 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 |

### 5.2.1.2 FAU_GEN.2 User Identity Association

**FAU_GEN.2.1**
For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

### 5.2.1.3 FAU_STG_EXT.1 Protected Audit Event Storage

**FAU_STG_EXT.1.1**
The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

**FAU_STG_EXT.1.2**
The TSF shall be able to store generated audit data on the TOE itself. In addition [

- *The TOE shall consist of a single standalone component that stores audit data locally*].

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

## 5.2.2 Cryptographic Support (FCS)

### 5.2.2.1 FCS_CKM.1 Cryptographic Key Generation

**FCS_CKM.1.1**
The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm: [

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

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

### 5.2.2.2 FCS_CKM.2 Cryptographic Key Establishment

**FCS_CKM.2.1**
The TSF shall **perform** cryptographic **key establishment** in accordance with a specified cryptographic key **establishment** method: [

- *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].~~

### 5.2.2.3 FCS_CKM.4 Cryptographic Key Destruction

**FCS_CKM.4.1**
The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

- *For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [zeroes]];*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [*
    - *logically addresses the storage location of the key and performs a* [*single overwrite consisting of* [*zeroes*]]]

that meets the following: *No Standard.*

### 5.2.2.4 FCS_COP.1/DataEncryption Cryptographic Operations (AES Data Encryption/Decryption)

**FCS_COP.1.1/DataEncryption**
The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [*CBC, CTR, GCM*] *mode* and cryptographic key sizes [*128 bits, 256 bits*] that meet the following: *AES as specified in ISO 18033-3,* [*CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772*].

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

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

- *RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits, 3072 bits]*
- *Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256-bits, 384-bits, 521-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,*
- *For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4*

].

### 5.2.2.6 FCS_COP.1/Hash Cryptographic Operations (Hash Algorithm)

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

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

**FCS_COP.1.1/KeyedHash**
The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512*] and cryptographic key sizes *[160 bits, 256 bits, 512 bits]* **and message digest sizes [*160, 256, 512*] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

### 5.2.2.8 FCS_RBG_EXT.1 Random Bit Generation

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

**FCS_RBG_EXT.1.2**
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [ *[2]* platform-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.

### 5.2.2.9 FCS_SSHS_EXT.1 SSH Server Protocol

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

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

**FCS_SSHS_EXT.1.3**
The TSF shall ensure that, as described in RFC 4253, packets greater than *[262,144]* bytes in an SSH transport connection are dropped.

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

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

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

**FCS_SSHS_EXT.1.7**
The TSF shall ensure that [*ecdh-sha2-nistp256*] and [*ecdh-sha2-nistp384, ecdh-sha2-nistp521*] are the only allowed key exchange methods used for the SSH protocol.

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

### 5.2.2.10 FCS_TLSC_EXT.1 TLS Client Protocol without Mutual Authentication

**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_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*

] *and no other ciphersuites.*

**FCS_TLSC_EXT.1.2**
The TSF shall verify that the presented identifier matches [*the reference identifier per RFC 6125 section 6, and no other attribute types*].

**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*].

**FCS_TLSC_EXT.1.4**
The TSF shall [*present the Supported Elliptic Curves/Supported Groups Extension with the following curves/groups*: [*secp256r1, secp384r1*] *and no other curves/groups*] in the Client Hello.

### 5.2.3 Identification and Authentication (FIA)

#### 5.2.3.1 FIA_AFL.1 Authentication Failure Management

**FIA_AFL.1.1**
The TSF shall detect when an Administrator configurable positive integer within *[1 and 65,535]* unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

**FIA_AFL.1.2**
When the defined number of unsuccessful authentication attempts has been met, the TSF shall [*prevent the offending Administrator from successfully establishing a remote session using any authentication method that involves a password until [the unlock account action] is taken by an Administrator; 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].*

#### 5.2.3.2 FIA_PMG_EXT.1 Password Management

**FIA_PMG_EXT.1.1**
The TSF shall provide the following password management capabilities for administrative passwords:

*a)* Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, [“~”, “`”, “-”, “ ”, “=”, “+”, “{”, “}”, “[”, “]”, “|”, “\”, “:”, “<”, “>”, “/”, “.”, “,”, “ ”]];

*b)* Minimum password length shall be configurable to between [*4*] and [*127*] characters.

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

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

- Display the warning banner in accordance with FTA_TAB.1;
- [*[responses to ping or ARP]*].

**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.

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

**FIA_UAU_EXT.2.1**
The TSF shall provide a local [*password-based*] authentication mechanism to perform local administrative user authentication.

#### 5.2.3.5 FIA_UAU.7.1 Protected Authentication Feedback

**FIA_UAU.7.1**
The TSF shall provide only *obscured feedback* to the administrative user while the authentication is in progress **at the local console**.

### 5.2.3.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certification path validation **supporting a minimum path length of three certificates** .
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [*a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3*].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsagefield.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

**FIA_X509_EXT.1.2/Rev**
The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

### 5.2.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

**FIA_X509_EXT.2.1**
The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [*TLS*] and [*no additional uses*].

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

**Application Note:** The application note associated with SFR has been updated as per TD0537.

## 5.2.4 Security Management (FMT)

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

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

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

**FMT_MTD.1.1/CoreData**
The TSF shall restrict the ability to manage the *TSF data to Security Administrators.*

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

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using [hash comparison] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- [
  - *Ability to configure the cryptographic functionality;*
  - *Ability to re-enable an Administrator account;*
  - *Ability to set the time which is used for time-stamps;*
  - *Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;*
  - *Ability to import X.509v3 certificates to the TOE's trust store*].

#### 5.2.4.4 FMT_SMR.2 Restrictions on Security Roles

**FMT_SMR.2.1**
The TSF shall maintain the roles:

- *Security Administrator.*

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

**FMT_SMR.2.3**
The TSF shall ensure that the conditions

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

are satisfied.

### 5.2.5 Protection of the TSF (FPT)

#### 5.2.5.1 FTP_APW_EXT.1 Protection of Administrator Passwords

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

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

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

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

#### 5.2.5.3 FPT_STM_EXT.1 Reliable Time Stamps

**FPT_STM_EXT.1.1**
The TSF shall be able to provide reliable time stamps for its own use.

**FPT_STM_EXT.1.2**
The TSF shall [*allow the Security Administrator to set the time*].

#### 5.2.5.4 FPT_TST_EXT.1 TSF Testing

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

- *Memory (RAM) walk*
- *File integrity verification*
- *Citrix FIPS Cryptographic Module tests:*
    - *Integrity check*
    - *Algorithm known answer tests].*

#### 5.2.5.5 FPT_TUD_EXT.1 Trusted Update

**FPT_TUD_EXT.1.1**
The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [*the most recently installed version of the TOE firmware/software*].

**FPT_TUD_EXT.1.2**
The TSF shall provide *Security Administrators* the ability to manually initiate updates to TOE firmware/software and [*no other update mechanism*].

**FPT_TUD_EXT.1.3**
The TSF shall provide means to authenticate firmware/software updates to the TOE using a [*published hash*] prior to installing those updates.

### 5.2.6 TOE Access (FTA)

#### 5.2.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

**FTA_SSL_EXT.1.1**
The TSF shall, for local interactive sessions, [

- *terminate the session*]

after a Security Administrator-specified time period of inactivity.

#### 5.2.6.2 FTA_SSL.3 TSF-initiated Termination

**FTA_SSL.3.1**
The TSF shall terminate **a remote** interactive session after a *Security Administrator-configurable time interval of session inactivity.*

#### 5.2.6.3 FTA_SSL.4 User-initiated Termination

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

#### 5.2.6.4 FTA_TAB.1 Default TOE Access Banner

**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.

### 5.2.7 Trusted Path/Channels (FTP)

#### 5.2.7.1 FTP_ITC.1 Inter-TSF Trusted Channel

**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, [*authentication server*]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **disclosure and detection of modification of the channel data**.

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

**FTP_ITC.1.3**
The TSF shall initiate communication via the trusted channel for *[export of audit logs to external audit server, authentication dialogue with authentication servers]*.

#### 5.2.7.2 FTP_TRP.1/Admin Trusted Path

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

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

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

## 5.3 TOE SFR Dependencies Rationale for SFRs

The PP and any relevant EPs/Modules/Packages contain(s) all the requirements claimed in this ST. As such, the dependencies are not applicable since the PP has been approved.

## 5.4 Security Assurance Requirements

The TOE assurance requirements for this ST are taken directly from the PP and any relevant EPs/Modules/Packages, which is/are derived from Common Criteria Version 3.1, Revision 5. The assurance requirements are summarized in the Table 11.

**Table 11 – Security Assurance Requirements**

| Assurance Class | Assurance Components | Component Description |
|---|---|---|
| Security Target | ASE_CLL.1 | Conformance claims |
| | ASE_ECD.1 | Extended components definition |
| | ASE_INT.1 | ST introduction |
| | ASE_OBJ.1 | Security objectives for the operational environment |
| | ASE_REQ.1 | Stated security requirements |
| | ASE_SPD.1 | Security problem definition |
| | ASE_TSS.1 | TOE summary specification |

| Assurance Class | Assurance Components | Component Description |
|---|---|---|
| Development | ADV_FSP.1 | Basic functional specification |
| Guidance Documents | AGD_OPE.1 | Operational user guidance |
|  | AGD_PRE.1 | Preparative Procedures |
| Life Cycle Support | ALC_CMC.1 | Labelling of the TOE |
|  | ALC_CMS.1 | TOE CM coverage |
| Tests | ATE_IND.1 | Independent testing – conformance |
| Vulnerability Assessment | AVA_VAN.1 | Vulnerability survey |

## 5.5 Assurance Measures

The TOE satisfied the identified assurance requirements. This section identifies the Assurance Measures applied by Citrix Systems Inc. to satisfy the assurance requirements. The following table lists the details.

**Table 12 – TOE Security Assurance Measures**

| SAR Component | How the SAR will be met |
|---|---|
| ADV_FSP.1 | The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). |
| AGD_OPE.1 | The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance. |
| AGD_PRE.1 | The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration. |
| ALC_CMC.1<br>ALC_CMS.1 | The Configuration Management (CM) documents describe how the consumer identifies the evaluated TOE. The CM documents identify the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked and how potential changes are incorporated. |
| ATE_IND.1 | Vendor will provide the TOE for testing. |
| AVA_VAN.1 | Vendor will provide the TOE for testing.<br>Vendor will provide a document identifying the list of software and hardware components. |

# 6 TOE Summary Specifications

This chapter identifies and describes how the Security Functional Requirements identifies above are met by the TOE.

**Table 13 – TOE Summary Specification SFR Description**

| Requirement | TSS Description |
|---|---|
| FAU_GEN.1<br>FAU_GEN.2 | The TOE generates audit records for commands executed by Administrators using the CLI and for other security-related events as shown in Table 9. In general terms the audit records include the date and time of the event, type of event (including the selected options in the case of administrator commands), subject identity (if applicable), the outcome (success or failure) of the event, and (if connecting remotely) the IP address of the relevant IT entity. Other details specific to each event are indicated in Table 12.<br>For the administrative task of managing cryptographic keys, the TOE identifies the relevant key in the following manner:<br>• KEK – There is only a single KEK, so any KEK operation implicitly identifies the KEK.<br>• X.509 Certificates – The administrator specifies a unique filename for the certificate.<br>• SSH Host Key – There is only a single SSH host key so any SSH host key operations implicitly identify the SSH host key.<br>• SSH User Public Keys – The full public key is logged when it is added or removed from the authorized keys file. |
| FAU_STG_EXT.1 | The TOE is a standalone product (and not comprised of multiple components). Audit records are stored on the TOE in /var/log in the ns.log file. Audit records are transmitted periodically to an external syslog server. The frequency of transfer of audit events will be dependent on the buffer being filled. Buffer size is set to about 5840 bytes. This is not a time-based logic. The buffer is maintained per core and whenever the buffer is filled beyond the threshold, the logs will be sent. The channel to the syslog server is protected using TLS as specified in FTP_ITC.1. When the connection to the syslog server is down, the audit records are stored locally. When the connection to the syslog server comes back up, the TOE will resume transmission of audit records to the syslog server; however, it does not transmit audit records generated while the connection was down.<br>The TOE stores 2,500KB of raw (uncompressed) audit records locally; however, actual physical storage space is less. The storage is segmented into the active ns.log file which is rotated when it reaches 100KB in size. The rotation process compresses the ns.log file and adds new audit records to a new ns.log file. If there are more than 25 compressed ns.log files, the TOE deletes the oldest file; effectively overwriting the previous audit records. These records are stored in an ACL protected directory allowing no unauthorized access. |
| FCS_CKM.1<br>FCS_CKM.2 | The TOE generates RSA 2048-bit and 3072-bit keys that are used as the SSH host key on the TOE. This key is generated according to FIPS 186-4. The TOE generates P-256, P-384, and P-521 ECDH/ECDSA keys to perform Elliptic curve-based key establishment in support of TLS and SSH as specified in SP 800-56A.<br>The TOE uses RSAES-PKCS1-v1_5 key transport as part of the TLS protocol. The TOE is the client/sender, so this operation does not involve RSA key generation.<br>The relevant NIST CAVP certificate numbers are listed in Table 2. |
| FCS_CKM.4 | Key destruction is found in Table 14 of Section 6.1. |
| FCS_COP.1/<br>DataEncryption | The TOE supports encryption and decryptions using AES-128 and AES-256 in CBC, CTR, and GCM modes. |

<table>
<tr><th>Requirement</th><th>TSS Description</th></tr>
<tr><td></td><td>AES-128 and AES-256 in CBC and CTR are used for SSH connectivity. AES-128 and AES-256 in CBC and GCM are used for TLS connectivity.<br>The relevant NIST CAVP certificate numbers are listed in Table 2.</td></tr>
<tr><td>FCS_COP.1/SigGen</td><td>The TOE supports cryptographic signature services using the RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 or 3072 bits or greater and Elliptic Curve Digital Signatures with P-256, P-384, and P-521 curves, meeting FIPS PUB 186-4.<br>These signature services are used in the TLS protocols as well as the SSH protocol (ssh-rsa).<br>The relevant NIST CAVP certificate numbers are listed in Table 2 .</td></tr>
<tr><td>FCS_COP.1/Hash</td><td>The TOE supports cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512. SHA-1 and SHA-256 are used in digital signatures. SHA-256 is used for update verification. SHA-256, SHA-384 and SHA-512 are used in the SSH KDF. SHA-1, SHA-256, and SHA-512 are used as HMAC primitives. Password hashing leverages PBKDF2.<br>The relevant NIST CAVP certificate numbers are listed in Table 2 .</td></tr>
<tr><td>FCS_COP.1/KeyedHash</td><td>The TOE supports the generation and verification of hash message authentication codes (HMACs) using HMAC-SHA-1, HMAC_SHA-256, and HMAC-SHA512. The details of each HMAC function are described below.
<table>
<tr><th></th><th>HMAC-SHA-1</th><th>HMAC-SHA-256</th><th>HMAC-SHA-512</th></tr>
<tr><td>Key Length</td><td>160 bits</td><td>256 bits</td><td>512 bits</td></tr>
<tr><td>Hash function</td><td>SHA-1</td><td>SHA-256</td><td>SHA-512</td></tr>
<tr><td>Block Size</td><td>512 bits</td><td>512 bits</td><td>1024 bits</td></tr>
<tr><td>Output MAC</td><td>160 bits</td><td>256 bits</td><td>512 bits</td></tr>
<tr><td>Uses</td><td>TLS KDF<br>TLS MAC</td><td>TLS KDF<br>SSH MAC</td><td>SSH MAC</td></tr>
</table>
The relevant NIST CAVP certificate numbers are listed in Table 2.</td></tr>
<tr><td>FCS_RBG_EXT.1</td><td>The TOE also generates random bits using two SP 800-90A CTR_DRBGs, both using AES-256.<br>• For management CPU/control plane, DRBG uses AES-CTR implementation with AES256<br>• For data plane/PE, DRBG uses AES-CTR implementation with AES256<br>• TLS is used in both Control Plane and Data plane and it uses respective DRBG to generate keys<br>• SSH is only used in control plane<br>The DRBG is used to generate keys and random bits for TLS. It is seeded using two third party hardware-based noise source that is assumed to produce at least 24 bits of entropy per 64-bit sample. These sources are fed into a single combiner. Entropy is collected until the requisite 256 bits of entropy is available. After that, the DRBG is seeded as required. The Citrix FIPS DRBG is used to generate keys for FCS_SSHS_EXT.1.<br>The relevant NIST CAVP certificate numbers are listed in Table 2.</td></tr>
<tr><td>FCS_SSHS_EXT.1</td><td>The TOE implements SSHv2 (compliant with RFCs 4251, 4252, 4253, 4254, 4344,</td></tr>
</table>

| Requirement | TSS Description |
|---|---|
| | 5656, and 6668) for administrators to make remote connections to access the CLI (as an alternative to the use of the local console). The TOE supports both ssh-rsa public key-based and password-based authentication methods for SSH.<br>The ssh-rsa private host key is stored in /nsconfig/ssh. When connecting over SSH, the ssh daemon looks up the relevant public key in the authorized_keys file. If a public key is present then it will be used for authentication, otherwise password-based authentication is used, passing the username and passphrase details to the PAM library to confirm their validity. If the authentication is successful, then the login process uses an exec system call to launch the CLI.<br>SSH packets larger than 256KB (262,144 bytes) are dropped by the TOE.<br>For SSH transport, the TOE uses aes128-cbc, aes256-cbc, aes128-ctr, or aes256-ctr to encrypt data. The data integrity algorithms used are hmac-sha2-256 and hmac-sha2-512.<br>The TOE uses only ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 as the SSH key exchange methods.<br>The TOE automatically rekeys the connection after 1 hour has elapsed or 1 GB of data has been encrypted with an encryption key. The TOE initiates the rekey upon reaching either threshold (whichever is reached first).<br>The TOE maintains a buffer for SSH packets received. The length of the received packets is calculated prior to any operation on the packet. If the packet length exceeds the maximum length supported by the TOE, the packet is dropped.<br>No additional optional characteristics of SSH are implemented regarding any of the supported algorithms. |
| FCS_TLSC_EXT.1 | The TOE implements TLS versions 1.1 (RFC 4346) and 1.2 (RFC 5246) with the following ciphersuites:<br>• TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268<br>• TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268<br>• TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492<br>• TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492<br>• TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492<br>• TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492<br>• TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246<br>• TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246<br>• TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289<br>• TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289<br>• TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289<br>• TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289<br>• TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289<br>• TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289<br>• TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289<br>• TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289<br>• TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288<br>• TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288<br>The TOE automatically configures references identifiers based on the FQDN configured by the administrator to connect to the TLS server. When a FQDN has been configured, the TOE establishes reference identifiers of DNS-ID and CN-ID. When the |

| Requirement | TSS Description |
|---|---|
| | TOE compares the reference identifies to the identifiers in the presented certificate, it will consider the identifiers matching if they are an exact match or if the presented identifier exactly matches with the exception of a wildcard in the left most position matching the left most position of the reference identifier. The TOE will use the SAN(s) in the presented certificate if present. The TOE will only use the CN if the certificate does not contain any SANs.<br>The TOE does not support certificate pinning.<br>The TOE will not establish the connection if the server certificate is invalid, if the presented identifier does not match, or if the CRL cannot be retrieved.<br>The TOE presents only the following Elliptic Curves secp256r1, secp384r1. This is by default. No configuration is required. |
| FIA_AFL.1 | The TOE is capable of tracking authentication failures for each of the claimed authentication mechanisms (local, SSH).<br>The administrator can configure the maximum number of failed attempts using the CLI interface via the **set aaa parameter -maxloginAttempts <num> -failedLoginTimeout <minutes>** - **-failedLoginTimeout <minutes>** -**persistentLoginAttempts ENABLED** command. The configurable range is between 1 and 65,535 attempts (i.e. a 32-bit integer). When a user account has sequentially failed authentication the configured number of times, the account will be locked. If the **-failedLoginTimeout** is configured, then the user account will be unlocked when the specified number of seconds have elapsed since the locking mechanism was engaged. If the administrator is required to intervene to unlock an account, this is done using the CLI via the **unlock aaa user <username>** CLI command from the local console.<br>Irrespective of whether an administrator intervened or whether the elapsed time occurred, when a locked account is unlocked, the failure counter associated with that user is reset to 0. If a user succeeds at authenticating before the locking mechanism has been enabled, the failure counter is reset to 0.<br>If the lockout attempts is set to, for example, 5 attempts, then the user will be locked out after the 5th consecutive failed login attempt. This means that the 6th and subsequent attempts will fail to gain access to the TOE even if the credential being offered is correct. The TOE prevents a situation where all administrator accounts are locked out by allowing local access for accounts that are blocked from remotely authenticating to the TOE. |
| FIA_PMG_EXT.1 | The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", """, "+", "-", ".", "/", ":", ";", "<", "=", ">", "?", "[", "\", "]", "^", "_", "`", "{", "\|", "}", "~", and " ". The minimum password length is settable by the Authorized Administrator and can range from 4 to 126 characters (in CC configuration minimum must be set to 15 characters). |
| FIA_UIA_EXT.1<br>FIA_UAU_EXT.2 | Administrators access the TOE through the CLI, either using a local console or via a remote connection using SSH. Identification and authentication is required for administrators before access is given to any of the TOE functions except for the display of the warning banner (as in FTA_TAB.1) or responses to ping or ARP.<br>The local console supports username/password credentials. The SSH connection supports a username with a password, via an external AAA server, or SSH public key authentication. |
| FIA_UAU.7 | When a user enters their password at the local console then no characters are displayed on the console. |
| FIA_X509_EXT.1/Rev | The TOE performs X.509 certificate validation at the following points: |

| Requirement | TSS Description |
|---|---|
| FIA_X509_EXT.2 | • Authentication of server X.509 certificates received during TLS session establishment.<br>• When certificates are loaded into the TOE, such as when importing CAs, certificate responses.<br>In all scenarios, certificates are checked for several validation characteristics:<br>• If the certificate 'not After' date is in the past, then this is an expired certificate which is considered invalid.<br>• If the certificate 'not Before' date is in the future, then this certificate is not yet valid which is considered invalid<br>• The certificate chain must terminate with a trusted root CA certificate.<br>• Server certificates consumed by the TOE TLS client must have a 'serverAuthentication' extendedKeyUsage purpose.<br>A trusted root CA certificate is defined as any certificate loaded into the TOE trust store. All CA certificates must have, at a minimum, a basicConstraints extension with the CA flag set to TRUE.<br>Certificate revocation checking is performed using CRLs. The TOE verifies the CA certificate used to sign the CRL has the CRLsign key usage bit set. If this bit is not set, the TOE considers this CRL invalid. If the TOE is unable to establish the connection to determine the validity of a certificate, the certificate shall be rejected. This check is performed when a certificate is presented for authentication and against all certificates in the trust chain.<br>As X.509 certificates are not used for either trusted updates or firmware integrity self-tests, the code-signing purpose is not checked for in the extendedKeyUsage.<br>The TOE has a trust store where root CA and intermediate CA certificates can be stored. The trust store is not cached: if a certificate is deleted, it is immediately untrusted. If a certificate is added to the trust store, it is immediately trusted for its given scope. The TOE checks each presented certificate against each certificate chain stored on the TOE to determine validity. Access to the trust store is limited to the Security Administrator.<br>The X.509 certificates for each of the given scenarios are validated using the certificate path validation algorithm defined in RFC 5280, which can be summarized as follows:<br>• The public key algorithm and parameters are checked<br>• The current date/time is checked against the validity period<br>• Revocation status is checked<br>• Issuer name of X matches the subject name of X+1<br>• Name constraints are checked<br>• Policy OIDs are checked<br>• Policy constraints are checked; issuers are ensured to have CA signing bits<br>• Path length is checked<br>• Critical extensions are processed<br>• If, during the entire trust chain verification activity, any certificate under review fails a verification check, then the entire trust chain is deemed untrusted and the TLS connection is terminated. |
| FMT_MOF.1/<br>ManualUpdate | The TOE restricts the ability to perform manual software updates to the Security Administrator role. |

| Requirement | TSS Description |
|---|---|
| FMT_MTD.1/CoreData | The TOE does not allow administrators to perform any administrative actions prior to administrator login. Once the administrator has successfully been identified and authenticated, the TOE restricts the ability to manage TSF data to the Security Administrator role. If a user other than the Security Administrator attempts to access any TSF data, that access is actively denied by the product. |
| FMT_SMF.1 | The TOE allows Security Administrators the ability to manage the following functions:<br>• Ability to configure the access banner.<br>• Ability to configure the session inactivity time before session termination.<br>• Ability to update the TOE, and to verify the updates using hash comparison prior to installing those updates.<br>• Ability to configure the authentication failure parameters for FIA_AFL.1.<br>• Ability to configure the cryptographic functionality.<br>• Ability to re-enable an Administrator account.<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.<br>• Ability to import X.509v3 certificates to the TOE's trust store.<br>All management functions are available from both the locally and remotely via SSH CLI. Local access is provided by a console port located on the front of the TOE. This is accessed by directly connecting a serial console cable to the TOE. |
| FMT_SMR.2 | The TOE maintains the Security Administrator role. This role maps to the ADC System User role. The TOE also supports a superuser role; however, this role may not be used in the evaluated configuration. |
| FPT_SKP_EXT.1 | The TOE does not provide a CLI interface designed to permit the viewing of pre-shared keys, symmetric keys or private keys. The TOE does not utilize pre-shared keys. The TOE only stores symmetric keys in RAM and does not provide any interface for reading these keys. Private keys are protected from access by the use of file and API permissions. The filesystem permissions prevent administrators from reading the SSH host key. |
| FPT_APW_EXT.1 | The TOE does not store passwords in plaintext form and does not provide an interface to view passwords (plaintext or hashes). Administrator passwords leverage PBKDF2 and password strings contained in audit log entries are obscured with asterisks |
| FPT_TST_EXT.1 | The TOE automatically runs the following self-tests at power-up:<br>• Memory (RAM) walk: This test involves applying a memory walk algorithm to portions of memory to ensure that it is not corrupt.<br>• File integrity verification using CRC32 and kernel image verification using RSA on the SHA-512 signature.<br>• Citrix FIPS Cryptographic Module tests:<br>○ Integrity check: This is a MAC applied over the cryptographic module.<br>○ Algorithm known answer tests: These tests involve injecting a known input into the cryptographic module and comparing the results to a known output.<br>If any failures are detected during the Memory walk, the TOE will take the memory module out of service and log the error. The TOE will continue to operate if one memory module remains operational. |

| Requirement | TSS Description |
|---|---|
| | If any of the other self-tests fail, the TOE will enter an error state and not provide any cryptographic services.<br>The self-tests demonstrate the TOE is operating correctly, because the integrity checks verify the executable code has not been modified and the algorithm known answer tests verify the hardware executing the instructions is operating correctly. |
| FPT_STM_EXT.1 | The TOE hardware provides a system clock, which is used for timestamps in audit log entries, to measure periods of inactivity during local and remote administrator sessions in order to determine when an inactive session is to be terminated, determine if certificates are valid, and determine the time-based SSH rekeying threshold.<br>The Administrator must manually update the time to ensure accuracy of the system clock. |
| FPT_TUD_EXT.1 | An Authorized Administrator can determine the current version of the TOE using the **show version** command and the **show hardware** command to display the hardware model identifier. The version of the inactive firmware can be queried using the **show version -installedversion** command.<br>Updates to the TOE software are obtained by an Administrator by download from the Citrix website. Each update is accompanied by a hash value that is also published on the Citrix website: before applying any update, the administrator applies the OpenSSL hash tool on the appliance, using the SHA-256 hash function and verifying that the hash value obtained matches the value published for that item on the Citrix website. Provided that the hash value is correct the Administrator then applies the update. |
| FTA_SSL.3<br>FTA_SSL_EXT.1 | An Authorized Administrator can specify a maximum inactivity time period for both local and remote interactive sessions after which a session will be automatically terminated by the TOE. |
| FTA_SSL.4 | An Administrator can choose to terminate their own interactive session from the CLI at any time using the 'logout' (or 'exit' or ctrl-d) command. |
| FTA_TAB.1 | An Authorized Administrator can specify a banner message that is displayed at the beginning of each administrative user session, both local console and SSH CLI. |
| FTP_ITC.1 | The TOE uses trusted channels based on TLS v1.1 and v1.2 (see FCS_TLSC_EXT.1) to communicate with external authentication servers and remote audit servers. These channels protect the communications from unauthorized disclosure or modification. The TOE initiates the connections to both server types. |
| FTP_TRP.1/Admin | The trusted path used for remote administrator connections is provided using SSH (see FCS_SSHS_EXT.1). |

## 6.1 Cryptographic Key Destruction

The table below describes the key zeroization provided by the TOE and as referenced in FCS_CKM.4:

**Table 14 – Cryptographic Key Specification**

| Keys/CSPs | Purpose | Storage Location | Method of Zeroization |
|---|---|---|---|
| EC/FFC Diffie Hellman private key | Key exchange private keys in support of TLS and SSH | RAM | Overwritten with zeros at the end of session or upon power off/reboot. |
| EC/FFC Diffie Hellman public key | Key exchange public keys in support of TLS and SSH | RAM | Overwritten with zeros at the end of session or upon power off/reboot. |

| Keys/CSPs | Purpose | Storage Location | Method of Zeroization |
|---|---|---|---|
| SSH Private Key | SSH host private key used in server authentication | ACL protected directory | Overwritten with zeros when the zeroization command is issued. |
| SSH Public Key | SSH host public key used in server authentication | N/A public | Overwritten with zeros when the zeroization command is issued. |
| SSH Session Key | Encryption keys associated with the SSH protocol | RAM | Overwritten with zeros at the end of session or upon power off/reboot. |
| TLS Session Encryption Key | Encryption keys associated with the TLS protocol | RAM | Overwritten with zeros at the end of session or upon power off/reboot. |
| TLS Session Integrity Key | Integrity keys associated with the TLS protocol | RAM | Overwritten with zeros at the end of session or upon power off/reboot. |

# 7 Acronym Table

Acronyms should be included as an Appendix in each document.

**Table 15 – Acronyms**

| Acronym | Definition |
|---|---|
| **AES** | Advanced Encryption Standard |
| **CC** | Common Criteria |
| **CRL** | Certificate Revocation List |
| **IP** | Internet Protocol |
| **NDcPP** | Network Device Collaborative Protection Profile |
| **NIAP** | Nation Information Assurance Partnership |
| **OCSP** | Online Certificate Status Protocol |
| **OSP** | Organizational Security Policy |
| **PP** | Protection Profile |
| **RSA** | Rivest, Shamir, & Adleman |
| **SFR** | Security Functional Requirement |
| **SSH** | Secure Shell |
| **ST** | Security Target |
| **TD** | Technical Decision |
| **TOE** | Target of Evaluation |
| **TLS** | Transport Layer Security |
| **TSS** | TOE Summary Specification |
| **vND** | virtual Network Device |