Cloud Software Group

NetScaler Version 13.1

Security Target

Document Version: 2.4

Prepared for:

A division of
Cloud Software Group
851 Cypress Creek Road
Fort Lauderdale, FL 33309
United States of America

Phone: +1 954 267 3000
www.cloud.com

Prepared by:

Corsec Security, Inc.

12600 Fair Lakes Drive
Suite 210
Fairfax, VA 22003
United States of America

Phone: +1 703 267 6050
www.corsec.com

Table of Contents

List of Figures

List of Tables

1. Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the ST organization. The Target of Evaluation (TOE) is the Cloud Software Group (Cloud) NetScaler Version 13.1, referred to hereafter as the TOE throughout this document. The TOE optimizes delivery of applications over the internet and private networks.

1.1 Purpose

This ST is divided into ten sections, as follows:

- Introduction (Section 1) – Provides a brief summary of the ST contents and describes the organization of other sections within this document. It also provides an overview of the TOE security functionality and describes the physical and logical scope for the TOE as well as the ST and TOE references.
- Conformance Claims (Section 2) – Provides the identification of any Common Criteria (CC), Protection Profile (PP), and Evaluation Assurance Level (EAL) package claims. It also identifies whether the ST contains extended security requirements.
- Security Problem (Section 3) – Describes the threats, Organizational Security Policies (OSPs), and assumptions that pertain to the TOE and its environment.
- Security Objectives (Section 4) – Identifies the security objectives that are satisfied by the TOE and its environment.
- Extended Components (Section 5) – Identifies new components (extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs)) that are not included in CC Part 2 or CC Part 3.
- Security Assurance Requirements (Section 6) – Presents the SARs met by the TOE.
- Security Functional Requirements (Section 7) – Presents the SFRs met by the TOE.
- TOE Summary Specification (Section 8) – Describes the security functions provided by the TOE that satisfy the SFRs and objectives.
- Rationale (Section 9) – Presents the rationale for the SFR dependencies as to their consistency, completeness, and suitability.
- Acronyms and Terms (Section 10) – Defines the acronyms and terminology used within this ST.

1.2 Security Target and TOE References

Table 1 shows the ST and TOE references.

Table 1 – ST and TOE References

ST Title	Cloud Software Group, NetScaler Version 13.1 Security Target
ST Version	Version 2.4
ST Author	Corsec Security, Inc.
ST Publication Date	December 2, 2024
TOE Reference	NetScaler Version 13.1 Build 37.201

1.3 Product Overview

The Product Overview provides a high-level description of the NetScaler Version 13.1 that is the subject of the evaluation. The following section, TOE Overview, will provide the introduction to the parts of the overall product offering that are specifically being evaluated.

The TOE is a network device that accelerates application performance, enhances application availability with advanced Layer 4 – Layer 7 load balancing, secures mission-critical apps from attacks and lowers server expenses by offloading computationally intensive tasks. All these capabilities are combined into a single, integrated appliance for increased productivity, with lower overall total cost of ownership.

1.4 TOE Overview

The TOE Overview provides a context for the TOE evaluation by identifying the TOE type, describing the product, and defining the specific evaluated configuration.

The TOE is purpose-built networking appliances whose function is to improve the performance, security and resilience of applications delivered over the web. NetScaler 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 NetScaler 13.1 software running on the following:

- NetScaler physical platforms
  - 8900 FIPS
  - 9100 FIPS
  - 15000-50G FIPS
- Virtual platform
  - Virtual appliance (with NetScaler CM) on ESXI 7.0 running on a Dell PowerEdge R630 Server

This evaluation is limited to the security functionality defined in the SFRs.

1.5 TOE Environment

The TOE relies on non-TOE hardware and software for its essential operation. Though this hardware and software is necessary for the TOE's operation, it is not part of the TOE. The following non-TOE hardware and software is required for essential operation of the TOE:

- Management Workstation (SSH[1] client)
- Syslog Server
- CA Server
- LDAP[2] Server
- RADIUS[3] Server
- CRL[4] Responder/CRL Distributer
- Managed Switch

[1] SSH – Secure Shell
[2] LDAP – Lightweight Directory Access Protocol
[3] RADIUS – Remote Authentication Dial-In Services
[4] CRL – Certificate Revocation List

It is assumed that there will be no untrusted users or software on the TOE Server components. In addition, the TOE Server components are intended to be deployed in a physically secured cabinet, room, or data center with the appropriate level of physical access control and physical protection (e.g., badge access, fire control, locks, alarms, etc.).

1.6 TOE Description

This section primarily addresses the physical and logical components of the TOE that are included in the evaluation.

1.6.1 Physical Scope

Figure 1 illustrates how the TOE is deployed, tying together all the components of the TOE and the constituents of the TOE environment. The TOE is a hardware and software TOE.

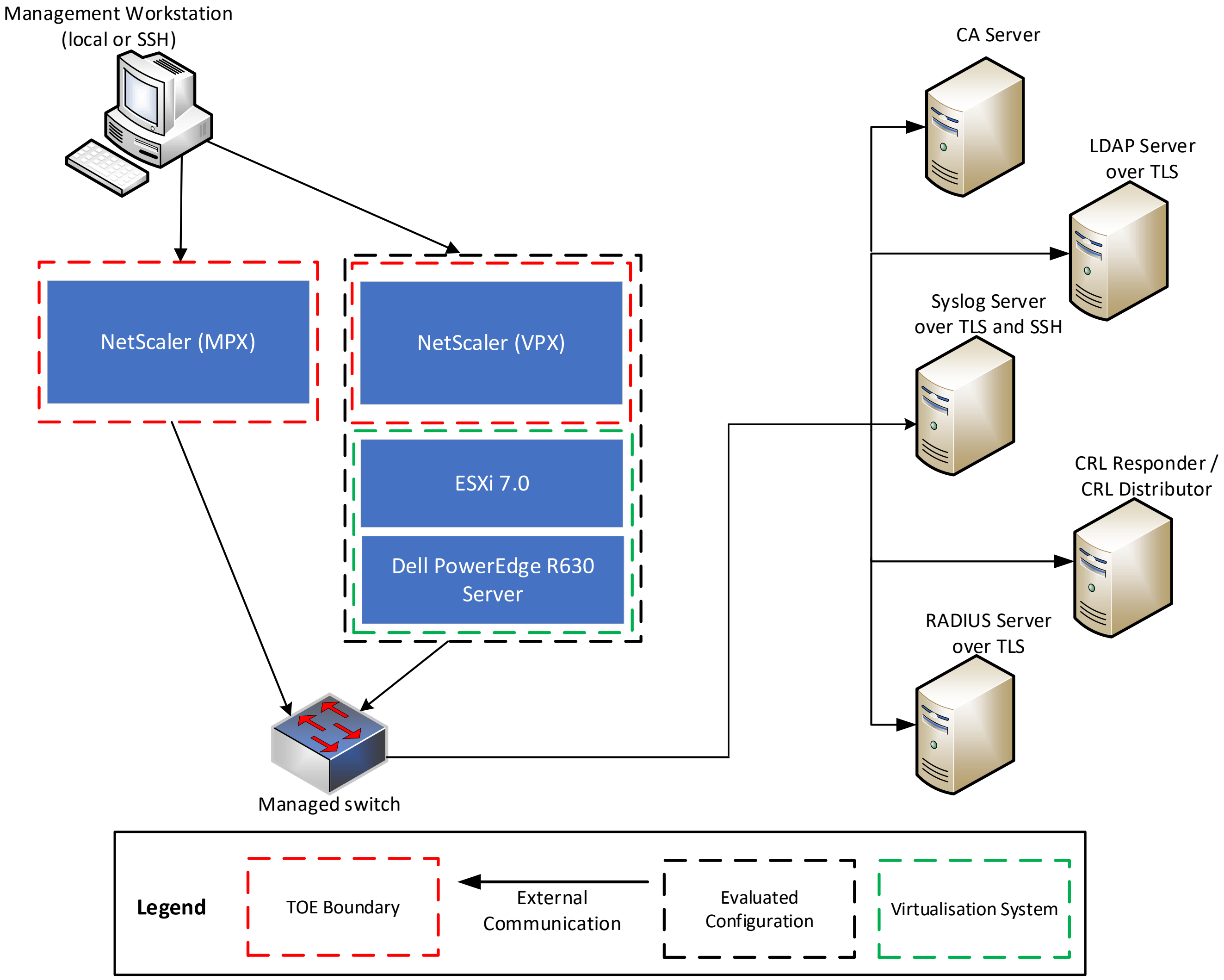

Figure 1 –TOE Deployment

1.6.1.1 TOE Hardware and Firmware

The TOE evaluated for configuration includes the following physical platforms: 8900 FIPS, 9100 FIPS, and 15000-50G FIPS. These are distributed to the customer via courier delivery. Each platform's processor is identified in the following table.

Table 2 –CPUs for Physical Platforms

Model	CPU
8900 FIPS	Intel Xeon E5-2620 v4 (Broadwell)
9100 FIPS	Intel Xeon Silver 4310T (Ice Lake)
15000-50G FIPS	Intel Xeon E5-2620 v4 (Broadwell)

The evaluated configuration also includes the VPX FIPS virtual platform, running on a VMware ESXi 7.0 hypervisor that is hosted by a Dell PowerEdge R630 server utilizing an Intel® Xeon E5-2680 v4 (Broadwell) processor. For distribution, the VPX FIPS virtual platform is downloaded by the customer.

FreeBSD 11.4 is the operating system on all the physical and virtual platforms.

1.6.1.2 Guidance Documentation

Table 3 lists the TOE Guidance Documentation needed to install, configure, and maintain the TOE. The Guidance Documentation can be downloaded off the Cloud Security Group Citrix website in PDF[5] format.

Table 3 – Guidance Documentation

Document Name	Description
Cloud Software Group NetScaler Version 13.1 Guidance Supplement v2.1	Contains configuration settings that should be applied to maintain the Common Criteria Evaluated Configuration.

1.6.2 Logical Scope

The TOE provides the security functions required by NDcPP v2.2e. The TOE is composed of the FreeBSD OS running directly on a physical appliance hardware and as a virtual appliance running on ESXi 7.0. The TOE is comprised of the following security classes, which are further described in sections 7 and 8 of this ST.

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

1.6.2.1 Security Audit

The TOE generates audit records for security relevant actions of the authorized administrators within the CLI. Audit records are sent periodically to an external Syslog server over a secure channel.

Audit records include:

- the date and time of the events
- type of events
- subject identity
- outcome of the events

[5] PDF – Portable Document Format

When the local storage space approaches its size limit, the TOE deletes the oldest file. Local audit records can be viewed by authorized administrators and are protected from unauthorized modification or deletion.

1.6.2.2 Cryptographic Support

The Cryptographic Support of the TSF function provides cryptographic functions to:

- secure sessions between the host machines connecting via SSH to the CLI of the TOE
- secure communications over TLS between the TOE and the external syslog server, RADIUS and LDAP servers

1.6.2.3 Identification and Authentication

The TOE provides functionality that requires administrators to verify their claimed identity. The Identification and Authentication TSF ensures that only legitimate administrators can gain access to the configuration settings and management settings of the TOE.

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

- Keyboard interactive authentication for external users with the `persistentLoginAttempts` parameter enabled in the system parameter.
- Password-based or public-key authentication for Security Administrators.
  - Security Administrators can set a minimum length for passwords of 4 to 127 characters.
- X.509v3 certificate-based authentication for remote devices.
  - The TOE only supports FQDN reference identifiers.

Device-level authentication allows the TOE to establish a secure communication channel with a remote endpoint. 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.

1.6.2.4 Security Management

The TOE provides CLI access for administrators to manage the security functions, configuration, and other features of the TOE. The Security Management function specifies the administrator-defined access for the management of the TOE. The following are TOE management functions provided via the local console port or remotely via SSH:

- Local console CLI administration
- Remote CLI administration via SSHv2
- Administrator authentication
- 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

1.6.2.5 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.

1.6.2.6 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.

1.6.2.7 Trusted Path/Channels

The TOE uses TLS to provide a trusted channel between itself and remote syslog, RADIUS and LDAP servers.

The TOE uses SSH to provide a trusted path between itself and remote administrators.

1.6.3 Excluded Features and Functionality

The following features and functionality are not part of the evaluated configuration of the TOE:

- Web Logging
- Application Firewall
- Global Server Load Balancing (GSLB)
- AAA-TM Authentication
- External authentication methods: Kerberos, TACACS+, SAML
- 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
- Content Inspection
- Connection Quality Analytics
- Adaptive TCP
- Forward Proxy
- Video Optimization
- URL Filtering
- SNMP[6]
- LOM[7] Port

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

- IP[8]v6
- NTP[9]-based updates to the time
- Use of superuser privileges except as described in [CCECG[10]]
- NetScaler GUI (HTTP/HTTPS), NetScaler Nitro API and ADM

1.6.4 Scope of Evaluation

The evaluation is limited in scope to the secure management features described in the *Collaborative Protection Profile for Network Devices* v2.2e, March 23, 2020, and detailed in section 1.6.2 of this document.

[6] SNMP – Simple Network Management Protocol

[7] LOM – Lights Out Management

[8] IP – Internet Protocol

[9] NTP – Network Time Protocol

[10] Common Criteria Evaluated Configuration Guide

2. Conformance Claims

This section provides the identification for any CC, PP, and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for CC and NDcPP[11] conformance claims can be found in section 9.1.

Table 4 – CC and PP Conformance

CC Identification and Conformance	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017; CC Part 2 extended; CC Part 3 conformant; PP claim to the collaborative *Protection Profile for Network Devices* v2.2e, March 23, 2020, conformant; and no interpretations apply to the claims made in this ST.
PP Identification	Exact Conformance[12] to the collaborative Protection Profile for Network Devices, v2.2e.

Table 5 - Relevant Technical Decisions

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD0800 – Updated NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	No	The evaluation does not include FCS_IPSEC_EXT.1. Therefore, this is not applicable.
TD0792 – NIT Technical Decision: FIA_PMG_EXT.1 – TSS EA not in line with SFR	Yes	
TD0790 – NIT Technical Decision: Clarification Required for testing IPv6	Yes	
TD0738 – NIT Technical Decision for Link to Allowed-With List	Yes	
TD0670 – NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	No	The evaluation does not include FCS_TLSC_EXT.2. Therefore, this is not applicable.
TD0639 – NIT Technical Decision for NTP MAC Keys	Yes	
TD0638 – NIT Technical Decision for Key Pair Generation for Authentication	No	The TOE is not a distributed TOE
TD0636 – NIT Technical Decision for Clarification of Public Key User Authentication for SSH	Yes	
TD0635 - NIT Technical Decision for TLS Server and Key Agreement Parameters	No	The evaluation does not include FCS_TLSS_EXT1. Therefore, this is not applicable.
TD0632 - NIT Technical Decision for Consistency with Time Data for vNDs[13]	Yes	
TD0631 - NIT Technical Decision for Clarification of public key authentication for SSH Server	Yes	
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	

[11] NDcPP - collaborative Protection Profile for Network Devices

[12] Exact Conformance is a type of Strict Conformance such that the set of SFRs and the Security Problem Definition/Objectives are exactly as presented within the accepted NDcPP without changes.

[13] virtual Network Device(s)

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
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_DTLSS_EXT.1.7 or FCS_TLSS_EXT1.4. 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 Decision 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.
TD0537 – NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	Yes	
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. Therefore, this is not applicable.
TD0527 – Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	

3. Security Problem Definition

This section describes the security aspects of the environment in which the TOE will be used and the way the TOE is expected to be employed. It provides the statement of the TOE security environment, which identifies and explains all the following:

- Known and presumed threats countered by either the TOE or by the security environment
- Organizational security policies with which the TOE must comply
- Assumptions about the secure usage of the TOE, including physical, personnel, and connectivity aspects

A Network Device has a network infrastructure role that it is designed to provide. In doing so, the Network Device communicates with other Network Devices and other network entities (i.e. entities not defined as Network Devices because they do not have an infrastructure role) over the network. At the same time, it must provide a minimal set of common security functionality expected by all Network Devices. The security problem to be addressed by a compliant Network Device is defined as this set of common security functionality that addresses the threats that are common to Network Devices, as opposed to those that might be targeting the specific functionality of a specific type of Network Device. The set of common security functionality addresses communication with the Network Device, both authorized and unauthorized, the ability to perform valid and secure updates, the ability to audit device activity, the ability to securely store and utilize device and Administrator credentials and data, and the ability to self-test critical device components for failures.

3.1 Threats

3.1.1 Threats to Security

This section identifies the threats to the IT[14] assets against which the TOE or security environment must provide protection. The threat agents are divided into two categories:

- Attackers who are not TOE users: These threat agents have public knowledge of how the TOE operates and are assumed to possess a low skill level, limited resources to alter TOE configuration settings or parameters, and no physical access to the TOE.
- TOE administrative users: These threat agents have extensive knowledge of how the TOE operates and are assumed to possess a high skill level, moderate resources to alter TOE configuration settings or parameters, and physical access to the TOE. (It is assumed that TOE administrative users will operate in a trusted manner.)

Both are assumed to have a low level of motivation. The IT assets requiring protection are the TSF and user data saved on the TOE. Removal, diminution, and mitigation of the threats are achieved through the objectives identified in section 4.

The threats for the Network Device are grouped according to functional areas of the device in the following subsections. The description of each threat is then followed by a rationale describing how it is addressed by the SFRs in section 7.

[14] IT – Information Technology

3.1.1.1 Communications with the Network Device

A Network Device communicates with other Network Devices and other network entities. The endpoints of this communication can be geographically and logically distant and may pass through a variety of other systems. The intermediate systems may be untrusted providing an opportunity for unauthorized communication with the Network Device or for authorized communication to be compromised. The security functionality of the Network Device must be able to protect any critical network traffic (administration traffic, authentication traffic, audit traffic, etc.). The communication with the Network Device falls into two categories: authorized communication and unauthorized communication.

Authorized communication includes network traffic allowable by policy destined to and originating from the Network Device as it was designed and intended. This includes critical network traffic, such as Network Device administration and communication with an authentication or audit logging server, which requires a secure channel to protect the communication. The security functionality of the Network Device includes the capability to ensure that only authorized communications are allowed and the capability to provide a secure channel for critical network traffic. Any other communication with the Network Device is considered unauthorized communication. (Network traffic traversing the Network Device but not ultimately destined for the device, e.g. packets that are being routed, are not considered to be 'communications with the Network Device' – cf. A.NO_THRU_TRAFFIC_PROTECTION in section 3.2.3.)

The primary threats to Network Device communications addressed in this ST focus on an external, unauthorized entity attempting to access, modify, or otherwise disclose the critical network traffic. A poor choice of cryptographic algorithms or the use of non-standardized tunnelling protocols along with weak Administrator credentials, such as an easily guessable password or use of a default password, will allow a threat agent unauthorized access to the device. Weak or no cryptography provides little to no protection of the traffic allowing a threat agent to read, manipulate and/or control the critical data with little effort. Non-standardized tunnelling protocols not only limit the interoperability of the device but lack the assurance and confidence standardization provides through peer review.

3.1.1.2 T.UNAUTHORIZED_ADMISTRATOR_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.

3.1.1.3 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

3.1.1.4 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.

3.1.1.5 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

Updating Network Device software and firmware is necessary to ensure that the security functionality of the Network Device is maintained. The source and content of an update to be applied must be validated by cryptographic means; otherwise, an invalid source can write their own firmware or software updates that circumvents the security functionality of the Network Device. Methods of validating the source and content of a software or firmware update by cryptographic means typically involve cryptographic signature schemes where hashes of the updates are digitally signed.

Unpatched versions of software or firmware leave the Network Device susceptible to threat agents attempting to circumvent the security functionality using known vulnerabilities. Non-validated updates or updates validated using non-secure or weak cryptography leave the updated software or firmware vulnerable to threat agents attempting to modify the software or firmware to their advantage.

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

Auditing of Network Device activities is a valuable tool for Administrators to monitor the status of the device. It provides the means for Administrator accountability, security functionality activity reporting, reconstruction of events, and problem analysis. Processing performed in response to device activities may give indications of a failure or compromise of the security functionality. When indications of activity that impact the security functionality are not generated and monitored, it is possible for such activities to occur without Administrator awareness. Further, if records are not generated and retained, reconstruction of the network and the ability to understand the extent of any compromise could be negatively affected. Additional concerns are the protection of the audit data that is recorded from alteration or unauthorized deletion. This could occur within the TOE, or while the audit data is in transit to an external storage device.

Note that the NDcPP requires that the Network Device generate the audit data and has the capability to send the audit data to a trusted network entity (e.g., a Syslog server).

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

A Network Device contains data and credentials which must be securely stored and must appropriately restrict access to authorized entities. Examples include the device firmware, software, configuration authentication credentials for secure channels, and Administrator credentials. Device and Administrator keys, key material, and authentication credentials need to be protected from unauthorized disclosure and modification. Furthermore, the security functionality of the device needs to require default authentication credentials, such as Administrator passwords, be changed.

Lack of secure storage and improper handling of credentials and data, such as unencrypted credentials inside configuration files or access to secure channel session keys, can allow an attacker to not only gain access to the Network Device, but also compromise the security of the network through seemingly authorized modifications to configuration or though man-in-the-middle attacks. These attacks allow an unauthorized entity to gain access and perform administrative functions using the Security Administrator's credentials and to intercept all traffic as an authorized endpoint. This results in difficulty in detection of security compromise and in reconstruction of the network, potentially allowing continued unauthorized access to Administrator and device 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 include 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

Security mechanisms of the Network device generally build up from roots of trust to more complex sets of mechanisms. Failures could result in a compromise to the security functionality of the device. A Network Device self-testing its security critical components at both start-up and during run-time ensures the reliability of the device's security functionality.

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

This section describes the assumptions made in identification of the threats and security requirements for Network Devices. The Network Device is not expected to provide assurance in any of these areas, and as a result, requirements are not included to mitigate the threats associated.

3.2.1 Assumptions for the TOE

3.2.1.1 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 NDcPP does not include any requirements on physical tamper protection or other physical attack mitigations. The NDcPP 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.

[OE.PHYSICAL]

3.2.1.2 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 computing platform for general purpose applications (unrelated to networking functionality).

vIf a virtual TOE evaluated as a pND[15], following Case 2 vND[16]s as specified in Section 1.2, the VS[17] 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.

[OE.NO_GENERAL_PURPOSE]

3.2.1.3 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 NDcPP. It is assumed that this protection will be covered by cPPs[18] and PP-Modules for particular types of Network Devices (e.g., firewall).

[OE.NO_THRU_TRAFFIC_PROTECTION]

[15] pND – physical Network Device
[16] vND – virtual Network Device
[17] VS – Virtual Server
[18] cPP - collaborative Protection Profile

3.2.1.4 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).

[OE.TRUSTED_ADMIN]

3.2.1.5 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.

[OE.UPDATES]

3.2.1.6 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.

[OE.ADMIN_CREDENTIALS_SECURE]

3.2.1.7 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.

[OE.RESIDUAL_INFORMATION]

3.2.1.8 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.

[OE.TRUSTED_ADMIN]

3.2.1.9 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.

[OE.UPDATES]

3.2.1.10 A.VS_ISOLATION (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.

[OE.VM_CONFIGURATION]

3.2.1.11 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.

[OE.VM_CONFIGURATION]

3.3 Organizational Security Policy

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. The description of each policy is then followed by a rationale describing how it is addressed by the SFRs in section 7.

3.3.1 OSPs for the TOE

3.3.1.1 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 section identifies security objectives for the TOE and its Operational Environment in terms of the security objectives for Network Devices. The Network Device is not expected to provide assurance in any of these areas, and as a result, requirements are not included to mitigate the threats associated.

4.1 Security Objectives for the TOE and its Operational Environment

4.1.1 Security Objectives for the TOE

There are no security objectives for the TOE.

4.1.2 Security Objectives for the Operational Environment

The following subsections describe objectives for the Operational Environment.

4.1.2.1 OE.PHYSICAL

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

4.1.2.2 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 its own VM, and does not include other VMs or the VS.

4.1.2.3 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.

4.1.2.4 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.

4.1.2.5 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.

4.1.2.6 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.

4.1.2.7 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.

4.1.2.8 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. Extended Components

This section defines the extended SFRs, and extended SARs met by the TOE.

5.1 Extended TOE Security Functional Components

All the extended requirements in this ST have been drawn from the NDcPP v2.2e. The NDcPP v2.2e defines the following extended SFRs, and since they are not redefined in this ST, Appendix C of the NDcPP v2.2e should be consulted for more information about those CC extensions.

Table 6 identifies all extended SFRs implemented by the TOE, and includes the Technical Decisions (TDs) that have modified each SFR.

Table 6 – Extended TOE Security Functional Requirements

SFR	Source	Relevant TDs
FAU_STG_EXT.1	NDcPP v2.2e	TD0592
FCS_RBG_EXT.1	NDcPP v2.2e	TD0581
FCS_TLSC_EXT.1	NDcPP v2.2e	TD0631, TD0790
FCS_SSHC_EXT.1	NDcPP v2.2e	TD0636
FCS_SSHS_EXT.1	NDcPP v2.2e	TD0631
FAU_GEN.1	NDcPP v2.2e	TD0592, TD0563
FAU_GEN.2	NDcPP v2.2e	TD0592, TD0563
FCS_CKM.1	NDcPP v2.2e	TD0581, TD0580
FCS_CKM.2	NDcPP v2.2e	TD0581, TD0580
FCS_CKM.4	NDcPP v2.2e	TD0581
FCS_COP.1/DataEncryption	NDcPP v2.2e	TD0581
FCS_COP.1/SigGen	NDcPP v2.2e	TD0581
FCS_COP.1/Hash	NDcPP v2.2e	TD0631
FCS_COP.1/KeyedHash	NDcPP v2.2e	TD0631
FIA_AFL.1	NDcPP v2.2e	TD0571, TD0570
FIA_PMG_EXT.1	NDcPP v2.2e	TD0792
FIA_X509_EXT.1/Rev	NDcPP v2.2e	TD0527
FIA_X509_EXT.2	NDcPP v2.2e	TD0527, TD0537
FPT_STM_EXT.1	NDcPP v2.2e	TD0563
FPT_TUD_EXT.1	NDcPP v2.2e	TD0536
FTP_ITC.1	NDcPP v2.2e	TD0572

5.2 Extended TOE Security Assurance Components

There are no extended TOE Security Assurance Components.

6. Security Assurance Requirements

This ST identifies the SARs to frame the extent to which the evaluator assesses the documentation applicable for the evaluation and performs independent testing.

This section lists the set of SARs from CC Part 3 that are required in evaluations against the NDcPP. Individual Evaluation Activities to be performed are specified in the *Supporting Document, Mandatory Technical Document, Evaluation Activities for Network Device cPP* [SD].

The general model for evaluation of TOEs against STs written to conform to the NDcPP is as follows: after the ST has been approved for evaluation, the ITSEF[19] will obtain the TOE, supporting environmental IT (if required), and the guidance documentation for the TOE. The ITSEF is expected to perform actions mandated by the Common Evaluation Methodology (CEM) for the ASE and ALC SARs. The ITSEF also performs the Evaluation Activities contained within the SD, which are intended to be an interpretation of the other CEM assurance requirements as they apply to the specific technology instantiated in the TOE. The Evaluation Activities that are captured in the SD also provide clarification as to what the developer needs to provide to demonstrate the TOE is compliant with the NDcPP.

The TOE security assurance requirements are identified in Table 7.

Table 7 – Security Assurance Requirements

Assurance Requirements	
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)	Labeling of the TOE (ALC_CMC.1)
	TOE CM[20] coverage (ALC_CMS.1)
Tests (ATE)	Independent testing – conformance (ATE_IND.1)
Vulnerability assessment (AVA)	Vulnerability survey (AVA_VAN.1)

[19] ITSEF – Information Technology Security Entrepreneurs Forum

[20] CM – Configuration Management

7. Security Functional 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.

7.1 Conventions

The conventions used in descriptions of the SFRs are as follows:

- Assignment: Indicated with italicized text surrounded by brackets (e.g., [*assignment*]).
- Refinement made in the PP: Indicated with bold text and strikethroughs (e.g., "**refinement**" or "~~refinement~~").
- Selection: Indicated with underlined text surrounded by brackets (e.g., [selection]).
- Assignment within a selection: Indicated with italicized and underlined text surrounded by brackets (e.g., [*assignment within a selection*]);
- Iteration: Indicated by adding a string starting with "/".
- Extended SFRs are identified by having a label 'EXT' at the end of the SFR name.
- Operations such as assignments and selections performed by the PP author are identified as shown above; however, they do not appear within brackets. This is done intentionally to delineate between selections or assignments made by the PP author and those made by the ST author. No refinements have been made by the ST author other than grammatical and formatting corrections, or those made in places where a table reference differs from that of the PP.

7.2 Security Functional Requirements

This section specifies the SFRs for the TOE and organizes the SFRs by CC class. Table 8 identifies all SFRs implemented by the TOE and indicates the ST operations made by the ST author performed on each requirement. Refinements made in the PP are also indicated.

Table 8 – TOE Security Functional Requirements

Class Name	Component Identification	Component Name
Security Audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_STG_EXT.1	Protected Audit Event Storage
Cryptographic Support	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_SSHC_EXT.1	SSH Client Protocol

Class Name	Component Identification	Component Name
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_TLSC_EXT.1	TLS Client Protocol
Identification and 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	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
Security Management	FMT_MOF.1/ManualUpdate	Management of security functions behavior
	FMT_MTD.1/CoreData	Management of TSF data
	FMT_MTD.1/CryptoKeys	Management of TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.2	Restrictions on Security Roles
Protection of the TSF	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric, and private keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF testing
	FPT_TUD_EXT.1	Trusted Update
TOE Access	FTA_SSL_EXT.1	TSF-initiated session locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE access banners
Trusted Path/Channels	FTP_ITC.1	Inter-TSF Trust Channel
	FTP_TRP.1/Admin	Trusted path (Refinement)

7.2.1 Class FAU: Security Audit

FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1

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

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

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

c. *All administrative actions comprising:*

- *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *Resetting passwords (name of related user account shall be logged).*

- *[no other actions];*

d. *Specifically defined auditable events listed in Table 9.*

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 NDcPP/ST, *information specified in column three of Table 9.*

Table 9 – 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_SSHC_EXT.1	• Failure to establish an SSH session	Reason for failure.
FCS_SSHS_EXT.1	• Failure to establish an SSH session	Reason for failure.
FCS_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 the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1	• Unsuccessful attempt to validate a certificate • Any Addition, replacement, or removal of trust anchors in the TOE's trust store	• Reason for failure of certification validation • 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_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FMT_SMR.2	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_STM_EXT.1	Discontinuous changes to time – either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FTA_SSL_EXT.1	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempts.
FTP_TRP.1/Admin	• Initiation of the trusted path. • Termination of the trusted path. • Failure of the trusted path functions.	None.

FAU_GEN.2 User identity association

FAU_GEN.2.1

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

FAU_STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1.1

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

FAU_STG_EXT.1.2

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

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

FAU_STG_EXT.1.3

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

7.2.2 Class FCS: Cryptographic Support

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[21] PUB[22] 186-4, "Digital Signature Standard (DSS)", Appendix B.3;*
- *ECC[23] 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;*

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

FCS_CKM.2 Cryptographic Key Establishment

FCS_CKM.2.1

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

- *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*~~].

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.

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

FCS_COP.1.1/DataEncryption

The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [*CBC, CTR, GCM*] *mode* and cryptographic key sizes [*128 bits, 256 bits*] that meet the following: *AES as specified in ISO[24] 18033-3*, [*CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772*].

[21] FIPS – Federal Information Processing Standard

[22] PUB – Publication

[23] ECC – Elliptic Curve Cryptography

[24] ISO – International Organization for Standardization

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*[25] *#1 v2.1 Signature Schemes RSASSA*[26]*-PSS*[27] *and/or RSASSA-PKCS1v1 5; ISO/IEC*[28] *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*

].

FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1/Hash

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

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

FCS_COP.1.1/KeyedHash

The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [*HMAC-SHA-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*[29] *Algorithm 2"*.

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 [*Hash_DRBG (any), CTR_DRBG (AES)*].

FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [*[1]* 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.

[25] PKCS – Public Key Cryptography Standard

[26] RSASSA – RSA Signature Scheme with Appendix

[27] PSS – Probabilistic Signature Scheme

[28] IEC – International Electrotechnical Commission

[29] MAC – Message Authentication Code

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, 8303 section 3.1, 8332*].

FCS_SSHS_EXT.1.2

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

FCS_SSHS_EXT.1.3

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

FCS_SSHS_EXT.1.4

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

FCS_SSHS_EXT.1.5

The TSF shall ensure that the SSH public-key based authentication implementation uses [*ssh-rsa, rsa-sha2-256, rsa-sha2-512*] 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.

FCS_SSHC_EXT.1 SSH Client Protocol

FCS_SSHC_EXT.1.1

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

FCS_SSHC_EXT.1.2

The TSF shall ensure that the SSH protocol implementation supports the following user authentication methods as described in RFC 4252: public key-based, [*none*].

FCS_SSHC_EXT.1.3

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

FCS_SSHC_EXT.1.4

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

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

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

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

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

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

FCS_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 RFC3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC3268*
- *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 RFC5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC5289*
- *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*].

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, secp521r1*] and no other curves/groups] in the Client Hello.

7.2.3 Class FIA: Identification and Authentication

FIA_AFL.1 Authentication Failure Management (Refinement)

FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within [*1 to 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 a local Administrator; prevent the offending remote Administrator from successfully authenticating until an Administrator defined time period has elapsed].*

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.

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.

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.

FIA_UAU.7 Protected Authentication Feedback

FIA_UAU.7.1

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

FIA_X509_EXT.1/Rev X.509 Certificate Validation

FIA_X509_EXT.1.1/Rev

The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.
- The certificate path must terminate with a trusted CA[30] certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [*a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3*].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

FIA_X509_EXT.1.2

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

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

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

7.2.4 Class FMT: Security Management

FMT_MOF.1/ManualUpdate Management of Security Functions Behavior

FMT_MOF.1.1/ManualUpdate

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

FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.1.1

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

[30] CA – Certificate Authority

FMT_MTD.1/CryptoKeys Management of TSF Data

FMT_MTD.1.1

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

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using [hash comparison] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- [
  - *Ability to manage the cryptographic keys;*
  - *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;*
  - *Ability to manage the trusted public keys database*].

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.

7.2.5 Class FPT: Protection of the TSF

FPT_APW_EXT.1 Protection of Administrator Passwords

FPT_APW_EXT.1.1

The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2

The TSF shall prevent the reading of plaintext passwords.

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

FPT_SKP_EXT.1.1

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

FPT_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*].

FPT_TST_EXT.1 TSF Testing (Extended)

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: [*list of self-tests run by the TSF*]: [

- *Memory (RAM) walk*
- *File integrity verification*
- *Cloud FIPS Cryptographic Module test:*
  - *Integrity check*
  - *Algorithm Known Answer Test (KAT)*[31]].

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.

7.2.6 Class FTA: TOE Access

FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1

The TSF shall, for local interactive sessions, [

- *terminate the session*]

after a Security Administrator-specified time period of inactivity.

FTA_SSL.3 TSF-initiated Termination (Refinement)

FTA_SSL.3.1

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

FTA_SSL.4 User-initiated Termination (Refinement)

FTA_SSL.4.1

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

FTA_TAB.1 Default TOE Access Banners (Refinement)

[31] KAT – Known Answer Test

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.

7.2.7 Class FTP: Trusted Path/Channels

FTP_ITC.1 Inter-TSF Trusted Channel (Refinement)

FTP_ITC.1.1

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

FTP_TRP.1/Admin Trusted Path (Refinement)

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.*

8. TOE Summary Specification

This section presents information to detail how the TOE meets the functional requirements described in previous sections of this ST.

8.1 TOE Security Functionality

Each of the security requirements and the associated descriptions correspond to the security functions. Hence, each function is described by how it specifically satisfies each of its related requirements. This serves to both describe the security functions and rationalize that the security functions satisfy the necessary requirements.

Table 10 – Mapping of TOE Security Functionality to Security Functional Requirements

Class Name	Component Identification	Component Name
Security Audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_STG_EXT.1	Protected Audit Event Storage
Cryptographic Support	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_SSHC_EXT.1	SSH Client Protocol
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_TLSC_EXT.1	TLS Client Protocol
Identification and 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	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
Security Management	FMT_MOF.1/ManualUpdate	Management of security functions behavior
	FMT_MTD.1/CoreData	Management of TSF data
	FMT_MTD.1/CryptoKeys	Management of TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.2	Restrictions on Security Roles
Protection of the TSF	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric, and private keys)
	FPT_STM_EXT.1	Reliable Time Stamps

Class Name	Component Identification	Component Name
	FPT_TST_EXT.1	TSF testing
	FPT_TUD_EXT.1	Trusted Update
TOE Access	FTA_SSL_EXT.1	TSF-initiated session locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE access banners
Trusted Path/Channels	FTP_ITC.1	Inter-TSF Trust Channel
	FTP_TRP.1/Admin	Trusted path (Refinement)

8.1.1 Security Audit

The Security Audit function provides the TOE with the functionality of generating audit records. As administrators manage and configure the TOE, their activities are tracked and recorded as audit records and are stored in the file system. The resulting audit records can be examined to determine which security relevant activities took place and who (i.e., which user) is responsible for those activities.

FAU_GEN.1, 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 8. 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. For the administrative task of managing cryptographic keys, the TOE identifies the relevant key in the following manner:

- KEK – There is only a single KEK, so any KEK operation implicitly identifies the KEK.
- X.509 Certificates – The administrator specifies a unique filename for the certificate.
- SSH Host Key – There is only a single SSH host key so any SSH host key operations implicitly identify the SSH host key.
- SSH User Public Key – The full public key is logged when it is added or removed from the authorized keys file.
- SSH User Private Key – The full private key is logged when it is added or removed from the public key file within the `/nsconfig/ssh` directory

FAU_STG_EXT.1

The TOE is a standalone product (and not comprised of multiple components). Audit records are stored on the TOE in the `/var/log` directory, in the `ns.log` file. System authorization details are stored in the `auth.log` file and bash script logs are stored in the `bash.log` file. SSH client logs with time details are stored in the `ssh_debug.log` file and build upgrade information is stored in the `notice.log` file. Real-time updates to `ns.log` are transmitted to an external syslog server as they happen. Updates to `ssh_debug.log`, `notice.log`, `auth.log` and `bash.log` are transmitted periodically to an external syslog server. The frequency of transfer of these updates is dependent on the buffer being filled. Buffer size is set to about 5840 bytes. 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 and SSH 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.

The maximum log space size of the TOE is a function of the size of the /var partition for each of the models, stored locally in an ACL protected directory, that does not allow unauthorized access. The storage is segmented into active ns.log, auth.log, bash.log, ssh_debug.log, and notice.log files.

If the space in the /var directory reaches the size of the /var partition, the checking of file sizes is halted, and new audit records are dropped until all the logs are purged to make room for new audit records and the checking process is restarted.

8.1.2 Cryptographic Support

Table 11 - CAVP Algorithm Certificate References

Algorithm	Description/Operation	Supported Mode / Standard	CAVP Cert. #	SFR
RSA	Signature Generation, Verification, and key transport	FIPS PUB 186-4	A3942, A3943, A3944	FCS_CKM.1 FCS_CKM.2 FCS_COP.1/SigGen FCS_COP.1/SigVer
ECDSA	EC Signature Services in support of SSH and TLS authentication	FIPS PUB 186-4	A3942, A3943, A3944	FCS_CKM.1 FCS_COP.1/SigGen FCS_COP.1/SigVer
AES	Encryption in support of TLS and SSH protocols	CTR, GCM, CBC ISO 18033-3, ISO 10116, ISO 19772	A3942, A3943, A3944	FCS_COP.1/DataEncryption
SHA	Cryptographic hashing services	ISO/IEC 10118-3:2004	A3942, A3943, A3944	FCS_COP.1/Hash
HMAC	Keyed hashing services	ISO/IEC 9797-2:2011	A3942, A3943, A3944	FCS_COP.1/KeyedHash
DRBG	Random number generation	ISO/IEC 18031:2011	A3942, A3943	FCS_RBG_EXT.1
KAS ECC	Key agreement	NIST SP 800-56A Revision 3	A3942, A3943, A3944	FCS_CKM.2

FCS_CKM.1, FCS_CKM.2

The TOE generates 2048-bit and 3072-bit RSA keys (as per FIPS 186-4) that are used as the SSH host key on the TOE. The TOE also generates 2048-bit and 3072-bit RSA keys (as per FIPS 186-4) that are used as the SSH client keys on the TOE to send audit logs using SCP. 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 Rev 3.

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. The TOE also uses RSAES-PKCS1-v1_5 for key establishment.

The relevant NIST CAVP certificate numbers are listed in Table 11.

FCS_CKM.4

Key destruction is described in the following table.

Table 12 - Cryptographic Key Specification

Key/CSP	Purpose	Storage	Method of Zeroization
EC/FFC Diffie Hellman private key	Key exchange private key in support of TLS and SSH	Plaintext in RAM	Overwritten with zeros at the end of session or upon power off/reboot.
EC/FFC Diffie Hellman public key	Key exchange public key in support of TLS and SSH	Plaintext in RAM	Overwritten with zeros at the end of session or upon power off/reboot.
SSH Server Private Key	SSH host private key used in server authentication	Plaintext in filesystem	Overwritten with zeros when the zeroization command is issued.
SSH Server Public Key	SSH host public key used in server authentication	Plaintext public	Overwritten with zeros when the zeroization command is issued.
SSH Client Private Key	SSH host private key used in client authentication for copying audit log date using SCP	Plaintext in filesystem	Overwritten with zeros when the zeroization command is issued.
SSH Client Public Key	SSH host public key used in client authentication for copying audit log date using SCP	Plaintext public	Overwritten with zeros when the zeroization command is issued.
SSH Session Key	Encryption key associated with the SSH protocol	Plaintext in RAM	Overwritten with zeros at the end of session or upon power off/reboot.
TLS Session Encryption Key	Encryption key associated with the TLS protocol	Plaintext in RAM	Overwritten with zeros at the end of session or upon power off/reboot.
TLS Session Integrity Key	Integrity key associated with the TLS protocol	Plaintext in RAM	Overwritten with zeros at the end of session or upon power off/reboot
Key-Encryption-Key (KEK)	Encrypt and protect sensitive data	Plaintext in filesystem	Overwritten with zeros when the zeroization command is issued.

FCS_COP.1.1/DataEncryption

The TOE supports encryption and decryption using AES-128 and AES-256 in CBC, CTR, and GCM modes. AES-128 and AES-256 in CTR are used for SSH connectivity.

AES-128 and AES-256 in GCM and CBC are used for TLS connectivity.

KEK is used to encrypt and protect sensitive data, such as passwords, keys, and secrets.

The relevant NIST CAVP certificate numbers are listed in Table 11.

FCS_COP.1.1/SigGen

The TOE supports cryptographic signature services using the RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 or 3072 bits and Elliptic Curve Digital Signatures with P-256, P-384, and P-521 curves, meeting FIPS PUB 186-4.

These signature services are used in the TLS protocols as well as the SSH protocol (`ssh-rsa, ssh-rsa2-256, and ssh-rsa2-512`).

The relevant NIST CAVP certificate numbers are listed in Table 11.

FCS_COP.1/Hash

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-512 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.

The relevant NIST CAVP certificate numbers are listed in Table 11.

FCS_COP.1/KeyedHash

The TOE supports the generation and verification of Hashed Message Authentication Codes (HMACs) using HMAC-SHA-1, HMAC_SHA-256, and HMAC-SHA512. The details of each HMAC functions are described in the following table.

Table 13 - HMAC Functions

	HMAC-SHA-1	HMAC-SHA-256	HMAC-SHA-512
Key Length	160 bits	256 bits	512 bits
Hash function	SHA-1	SHA-256	SHA-512
Block Size	512 bits	512 bits	1024 bits
Output MAC	160 bits	256 bits	512 bits
Uses	TLS KDF TLS MAC	TLS KDF SSH MAC	SSH MAC

The relevant NIST CAVP certificate numbers are listed in Table 11.

FCS_RBG_EXT.1

For physical and virtual platforms, the TOE generates random bits using SP 800-90A CTR_DRBG, using AES-256 and SHA2-256 hash-based DRBG.

- For the management CPU/Control Plane, the DRBG uses an AES-CTR implementation with AES256.
- For the Data Plane/PE, the hash-based DRBG uses a SHA2-256.
- TLS is used for the Data Plane, and it uses the DRBG to generate keys.
- SSH is only used in the Control Plane.

In all instances, the entropy source is the NetScaler CPU Jitter Entropy Source. The entropy source collects entropic data until the requisite 256 bits of entropy are available. The entropy source uses a SHA3-256 hash function as the conditioning component, which is a vetted conditioning component according to SP800-90B. Additionally the CAVP certificate for the NetScaler CPU Jitter Entropy Source is A3513. After that, each DRBG is seeded as required. The following results are for the VPX, MPX-89xx, MPX-91xx and the 15xxx-50G (each min-entropy is the estimated bits of entropy per four bits of noise data):

Table 14 - Final Raw Min-Entropy Estimates

Entropy Result	VPX	MPX 8900	MPX 9100	MPX 15000-50G
Min-Entropy	3.446767	3.538870	3.666563	3.538870

As long as there is at least one bit of entropy per four bits of raw noise data, the entropy provided by each call to CPU Jitter entropy can be considered to contain full entropy. When the DRBG requests 256 bits of entropy for seeding, the function is called four times and returns 256 bits of entropy.

The entropy source within the TSF is SP800-90B and SP800-90C compliant.

The Control Plane DRBG is used to generate keys for FCS_SSHS_EXT.1.

The relevant NIST CAVP certificate numbers are listed in Table 11.

FCS_SSHS_EXT.1, FCS_SSHC_EXT.1

For the SSH Server and host-key algorithms, the TOE implements SSHv2 (compliant with RFCs 4251, 4252, 4253, 4254, 4344, 5656, 6668, 8303 section 3.1, and 8332) for administrators to make remote connections to access the CLI (as an alternative to the use of the local console). The TOE supports `ssh-rsa`, `rsa-sha2-256`, and `rsa-sha2-512` for public key-based authentication and password-based authentication for SSH.

For the SSH Client and user-key algorithms, the TOE implements SSHv2 (compliant with RFCs 4251, 4252, 4253, 4254, 4344, 5656, and 6668). The TOE supports `ssh-rsa` for public key-based authentication to the SSH server. The TOE SSH Client will accept external host keys of type `ssh-rsa` from the remote non-TOE SSH server that have been properly identified as a known host when configuring the SSH Server.

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.

SSH packets larger than 256 KB (262,144 bytes) are dropped by the TOE. For SSH transport, the TOE uses `aes128-`ctr or `aes256-ctr` to encrypt data. The data integrity algorithms used are `hmac-sha2-256` and `hmac-sha2-512`. The SSH Client also utilizes the same algorithms and key sizes for data encryption and data integrity.

The TOE uses only `ecdh-sha2-nistp256`, `ecdh-sha2-nistp384`, and `ecdh-sha2-nistp521` as the SSH key exchange methods. 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).

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.

The TOE ensures the SSH client authenticates the identity of the SSH server using a local database that pairs each host name with its associated public key.

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:

- 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

The TOE automatically configures references identifiers based on the FQDN[32] 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 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[33](s) in the presented certificate if present. The TOE will only use the CN[34] if the certificate does not contain any SANs.

The TOE does not support certificate pinning.

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.

The TOE presents the following Elliptic Curves secp256r1, secp384r1, and secp521r1.

The TOE sets up a single TLS connection to external services (LDAP, RADIUS, SYSLOG, etc.) when utilizing a single vserver configuration and acts as a central TLS processing engine.

8.1.3 Identification and Authentication

FIA_AFL.1

The TOE is capable of tracking password authentication failures for each of the claimed authentication mechanisms (local, SSH). The administrator can configure the maximum number of failed attempts using the CLI interface via

```
set aaa parameter -maxloginAttempts <value> -failedLoginTimeout <value> -
persistentLoginAttempts ENABLED
```

[32] Fully Qualified Domain Name
[33] Subject Alternative Name
[34] Common Name

The configurable range is between 1 and 65,535 attempts (i.e. a 16-bit integer). When a user account has sequentially failed authentication the configured number of times, the account will be locked. If –`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 `unlock aaa user <username>`.

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.

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 and 127 characters (in CC configuration minimum must be set to 15 characters).

FIA_UIA_EXT.1, 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 the display of the warning banner (as in FTA_TAB.1) or responses to ping or ARP. 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. If the credentials provided are correct, no errors are given, and the TOE no longer asks for the password. For SSH public key authentication, a successful connection to the TOE means a successful login.

FIA_UAU.7

When a user enters their password at the local console, no characters are displayed on the console.

FIA_X509_EXT.1, FIA_X509_EXT.2

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

- Authentication of server X.509 certificates received during TLS session establishment.
  - The TOE only supports FQDN reference identifiers.
- When certificates are loaded into the TOE, such as when importing CAs, certificate responses.

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

- If the certificate ‘not After’ date is in the past, then this is an expired certificate which is considered invalid.
- If the certificate ‘not Before’ date is in the future, then this certificate is not yet valid, which is considered invalid.
- The certificate chain must terminate with a trusted root CA certificate.
- Server certificates consumed by the TOE’s TLS client must have a `serverAuthentication extendedKeyUsage` attribute.

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.

Certificate revocation checking is performed using CRLs. The TOE verifies that the CA certificate used to sign the CRL has the `CRLsign` key usage bit set. If this bit is not set, the TOE will consider 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.

As X.509v3 certificates are not used for either trusted updates or firmware integrity self-tests, the code-signing purpose is not checked for in the `extendedKeyUsage` attribute.

The TOE has a trust store where root CA and intermediate CA certificates can be stored. The trust store is not cached: if a CA 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.

The X.509v3 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:

- The public key algorithm and parameters are checked
- The current date/time is checked against the validity period
- The revocation status is checked
- Issuer name of X matches the subject name of X+1
- Name constraints are checked
- Policy OIDs are checked
- Policy constraints are checked; issuers are ensured to have CA signing bits
- The path length is checked
- Critical extensions are processed
- 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

Certificates need to be added using the `add ssl certkey` command and bound on the particular vserver in use. The certificates bound on the vserver are used.

When configuring the CRL, auto refresh must be enabled to ensure the source of revocation information is a dynamically refreshed source.

Additional administrative configuration for the operating environment is available in the *Cloud Software Group NetScaler Version 13.1 Guidance Supplement*.

8.1.4 Security Management

FMT_MOF.1/ManualUpdate

The TOE restricts the ability to perform manual software updates to the Security Administrator role.

FMT_MTD.1/CoreData

The TOE does not allow administrators to perform any administrative actions prior to administrator login. Once an 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 denied.

FMT_MTD.1/CryptoKeys

The TOE restricts the ability to modify, delete, generate, or import cryptographic keys to the Security Administrator role.

FMT_SMF.1

The TOE allows Security Administrators the ability to manage the following functions:

- Ability to configure the access banner.
- Ability to configure the session inactivity time before session termination.
- Ability to update the TOE, and to verify the updates using hash comparison 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.
- Ability to manage the trusted public keys database

All management functions are available both locally and remotely via the 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, which maps to the NetScaler System User role. The TOE also supports a superuser role; however, this role must not be used in the evaluated configuration.

8.1.5 Protection of the TSF

FPT_SKP_EXT.1

The TOE does not provide an interface to permit the viewing of pre-shared keys, symmetric keys or private keys. The TOE does not utilize pre-shared keys. The TOE 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 TOE platform's 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. Administrator passwords leverage PBKDF2, and password strings present in audit log entries are obscured with asterisks.

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, determine the time-based CRL refresh, determine the time-based SSH-based audit log delivery is done periodically, and determine the time-based SSH rekeying threshold.

The Administrator must manually update the time to ensure accuracy of the system clock.

FPT_TST_EXT.1

The TOE automatically runs the following self-tests at power-up:

- Memory (RAM) walk: This test involves applying a memory walk algorithm to portions of memory to ensure that it is not corrupt.
- File integrity verification using CRC32 and kernel image verification using RSA on the SHA-512 signature.
- Cloud FIPS Cryptographic Module tests:
    - Integrity check: This is a MAC applied over the cryptographic module.
    - Algorithm known answer tests: These tests involve injecting a known input into the cryptographic module and comparing the results to a known output.

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.

If the module enters the critical error state due to a failure of the pre-operational integrity test, the module enters a critical error state and logs an error message. In this state, the boot sequence and entire system is halted. The only action available from this state is to reboot the module to trigger the re-execution of the integrity test. The error condition is considered to have been cleared if the module successfully passes the pre-operational integrity test. If the module continues to return to a halted state, the module is considered to be malfunctioning or compromised, and Cloud Customer Support must be contacted.

If the module enters the critical error state due to a failure of any of the conditional CASTs, cryptographic operations are halted, and the module inhibits all data output from the module. The module logs an error message and automatically reboots to clear the error state. Cloud Software Group must be contacted if this error occurs.

The successful completion or failure of the pre-operational self-tests and conditional CASTs (Conditional cryptographic algorithm self-tests) can be verified by checking the log files.

- Netscaler Control Plane Cryptographic Library (Cert. A3942) – Successful completion of the self-tests is indicated by "POST Success" in `/var/log/FIPS-post.log`. Failure is indicated by "POST Failed" in `/var/log/FIPS-post.log` (both messages indicate a critical error state).
- Netscaler Data Plane Cryptographic Library (Cert. A3943)– Successful completion of the self-tests is indicated by "FIPS POST Successful" in `/var/log/ns.log`. Failure is indicated by "FIPS Post Failed" in /var/log/ns.log (both messages indicate a critical error state).
- Intel Hardware Cryptographic Accelerator – (Cert. A3944) Successful completion of the self-tests is indicated by "FIPS POST Successful" in `/var/log/ns.log`. Failure is indicated by "FIPS Post Failed" in `/var/log/ns.log` (both messages indicate a critical error state).

If any of the remaining conditional self-tests fail, the module goes through a soft error state and the following message is displayed:

```
"Internal failure in SSL cert/key generation tool"
```

For these failures, the module returns to an operational state once the message is displayed (and the error is logged). The user may retry the service (which calls the conditional self-test again) or move to other operations. Successful completion of the conditional self-test is indicated by the absence of an error message.

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_TUD_EXT.1
An Authorized Administrator can determine the current version of the TOE using `show version` and `show hardware` to display the hardware model identifier. The version of the installed but inactive firmware can be queried using `show version -installedversion`.

Updates to the TOE software are downloaded from the Cloud website by an Administrator, with each update accompanied by a separate published hash. Before an update can be applied, the Administrator must invoke TOE capabilities to validate the software by checking the hash value. This is done by the admin invoking TOE capabilities through the CLI to generate a fresh hash value of the update and then comparing the newly-generated hash to the published hash. Only after visually confirming that the hashes are the same will the Administrator then apply the update; otherwise the Administrator will contact vendor support and halt the update process.

8.1.6 TOE Access

FTA_SSL_EXT.1, FTA_SSL.3
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. By default, sessions are terminated after 15 minutes of inactivity, but this value can be between 5 minutes and 24 hours.

FTA_SSL.4
An Administrator can choose to terminate their own interactive session from the CLI at any time using

```
logout / exit / ctrl-d
```

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.

8.1.7 Trusted Path/Channels

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 (RADIUS, LDAP) and both TLS v1.1 and v1.2 and SSH to communicate with remote audit servers (syslog). These channels protect the communications from unauthorized disclosure or modification. The TOE initiates the connections to both server types. If the trusted path to the logging channel is broken, an authorized administrator would need to reestablish the trusted path.

FTP_TRP.1/Admin
The trusted path used for remote administrator connections is provided using SSH (see FCS_SSHS_EXT.1).

9. Rationale

9.1 Conformance Claims Rationale

This Security Target extends Part 2 and conforms to Part 3 of the Common Criteria Standard for Information Technology Security Evaluations, Version 3.1 Revision 5. This ST conforms to the NDcPP v2.2e.

9.1.1 Variance Between the PP and this ST

There is no variance between the NDcPP and this ST.

9.1.2 Security Assurance Requirements Rationale

This ST claims exact conformance to the NDcPP, including the assurance requirements listed in Section 6 of the NDcPP.

9.1.3 Dependency Rationale

The SFRs in this Security Target represent the SFRs identified in the NDcPP v2.2e. As such, the NDcPP v2.2e SFR dependency rationale is deemed acceptable since the NDcPP itself has been validated.

10. Acronyms and Terms

Table 15 and Table 16 describe the acronyms and terms used throughout the document.

10.1 Acronyms

Table 15 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
API	Application Programming Interface
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CC	Common Criteria
CEM	Common Evaluation Methodology
CLI	Command Line Interface
CM	Configuration Management
CN	Common Name
cPP	collaborative Protection Profile
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CRL	Certificate Revocation List
DNS	Domain Name System
DRBG	Deterministic Random Bit Generator
EAL	Evaluation Assurance Level
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
ESXi	Elastic Sky X integrated
FIPS	Federal Information Processing Standard
FQDN	Fully Qualified Domain Name
GCM	Galois/Counter Mode
GUI	Graphical User Interface
HMAC	Hashed Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
ID	Identifier
IEC	International Electrotechnical Commission
IP	Internet Protocol
ISO	International Organization for Standardization
IT	Information Technology
ITSEF	Information Technology Security Entrepreneurs Forum
KAT	Known Answer Test
LDAP	Lightweight Directory Access Protocol
LOM	Lights Out Mangement
MAC	Message Authentication Code
ND	Network Device

Acronym	Definition
NDcPP	collaborative Protection Profile for Network Devices v2.2e
NIST	National Institute of Standards and Technology
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OSP	Organizational Security Policy
OU	Organizational Unit
PBKDF2	Password Based Key Derivation Function
PCT	Pairwise Consistency Test
PDF	Portable Document Format
PKCS	Public Key Cryptography Standard
PP	Protection Profile
PSS	Probabilistic Signature Scheme
PUB	Publication
RADIUS	Remote Authentication Dial-In Services
RAM	Random Access Memory
RBG	Random Bit Generator
RFC	Request For Comment
RSA	Rivest, Shamir, Adleman
RSASSA	RSA Signature Scheme with Appendix
SAN	Subject Alternative Name
SAR	Security Assurance Requirement
SD	Supporting Document
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SNMP	Simple Network Management Protocol
SP	Special Publication
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
VM	Virtual Machine
vND	Virtual Network Device
VS	Virtual Server

10.2 Terms

Table 16 – Terms

Name	Definition
Administrator	See Security Administrator.
Assurance	Grounds for confidence that a TOE meets the SFRs.
Security Administrator	The terms “Administrator” and “Security Administrator” are used interchangeably in this document.

Name	Definition
TSF Data	Data for the operation of the TSF upon which the enforcement of the requirements relies.

Prepared by:
Corsec Security, Inc.

12600 Fair Lakes Drive, Suite 210
Fairfax, VA 22003
United States of America

Phone: +1 703 267 6050
Email: info@corsec.com
http://www.corsec.com