SonicWall SMA v12.4
Security Target

Version 0.5
Sep 22, 2021

Table of Contents

Figures and Tables

1 Security Target Introduction

1.1 Security Target Reference

ST Title: SonicWall SMA v12.4 Security Target

ST Version: v0.5

ST Author: Cygnacom Solutions Inc.

ST Date: 09/22/2021

1.2 TOE Reference

TOE Developer: SonicWall

Evaluation Sponsor: SonicWall

TOE Identification: SonicWall Secure Mobile Access (SMA) v12.4.1

Table 1: TOE Platforms and Devices

Series	Platforms	Build
SonicWall Secure Mobile Access SMA1000 Series	SMA 6210	12.4.1-02451 [1]
	SMA 7210	
	SMA 8200v	

CC Identification: Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017.

PP Identification: collaborative Protection Profile for Network Devices, Version 2.2e, March 2020.

[1] Core build SMA 12.4.1-02451 with pform-hotfix-12.4.1-02559 and Connect Tunnel 12.4.1.939 was used in testing.

1.3 TOE Overview

1.3.1 *TOE Product Type*

The Target of Evaluation [TOE] is a Network Device as defined by the *collaborative Protection Profile for Network Devices v2.2e* [NDcPP]: "*A network device in the context of this cPP is a device composed of both hardware and software that is connected to the network and has an infrastructure role within the network".*

1.3.2 *TOE Usage*

The TOE is the SonicWall Secure Mobile Access (SMA) v12.4.1 which in the evaluated configuration includes SMA 6210, SMA 7210 appliances and SMA 8200v virtual appliance. SonicWall SMA is a unified secure access gateway that enables organizations to provide anytime, anywhere and any device access to corporate resources.

All SMA hardware appliances are shipped ready for immediate access through a Command Line Interface (CLI) and after basic network configuration through a web-based Appliance Management Console (AMC). Virtual appliance requires installation into hypervisor environment and supports configuration through AMC. To ensure secure use of the product, it must be appropriately configured prior to being put into a production environment as specified in the user guidance.

1.3.3 *TOE Security Functionality*

- Security Audit
    - Audit record generation for security-relevant events
    - Interoperability with a remote audit server
- Cryptographic Support
    - Validated cryptographic primitives
    - Destruction of cryptographic keys
    - Entropy generation
- Identification and Authentication
    - Authentication failure policies
    - Password management policies
    - Password and certificate based authentication
- Security Management
    - Local and remote administration
- Protection of the TOE Security Function (TSF)
    - Self-testing on power-up
    - Trusted update
- TOE Access
    - Session timeout and lockout
    - Access banner
- Trusted Path/Channels
    - Secure channel for remote administrators
    - Secure channel for communicating with authorized IT entities

1.4 TOE Description

The evaluated product name is SonicWall Secure Mobile Access (SMA) v12.4, and the evaluated version of the TOE is 12.4.1. The SonicWall SMA v12.4 functions as a remote access gateway operating as an intermediary device between end users and network resources residing on an internal network. The TOE provides multiple access methods for end users or client devices to remotely access an internal network resources from untrusted external networks. The SMA administrator configures policies comprised of security rules operating on users and targeting resources that must be satisfied in order to establish remote access.

The TOE is offered as SMA 6210 and SMA 7210 hardware appliances and SMA 8200v virtual appliance that are part of SMA1000 product line. The SMA 6210 and SMA 7210 are identical except for CPU, RAM, and SFP+ ports. The SMA 8200v is a virtual appliance designed to operate in a virtualization environment.

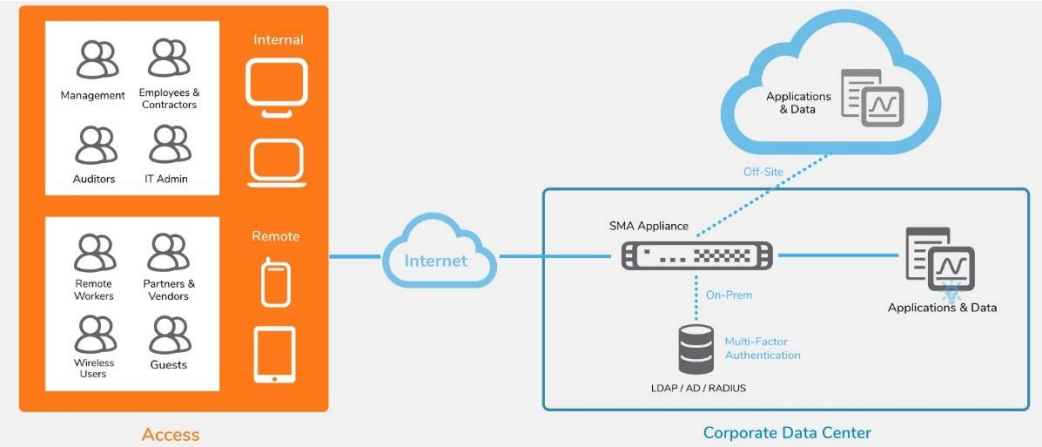

Figure 1: SMA Appliance

1.4.1 *TOE Architecture*

The architecture of each hardware appliance consists of generic hardware that supports physical network connections, memory, storage and computing resources. In case of the virtual appliance, the hardware is abstracted by the virtualization environment. The software includes the operating system and the SMA application software. The application software implements End User and Control & Configuration planes. Control & Configuration functionality includes all Security Functionality claimed in this document including administration, while End User is the gateway functionality that implements access to the internal network resource. While hardware varies between the appliance models, the software and End User and Control & Configuration is consistent across all evaluated appliances.

There are numerous open source and proprietary components packaged in the software, but only those relevant to the TOE's SF are presented in this reference architecture (Figure 2: TOE Architecture) for simplicity.

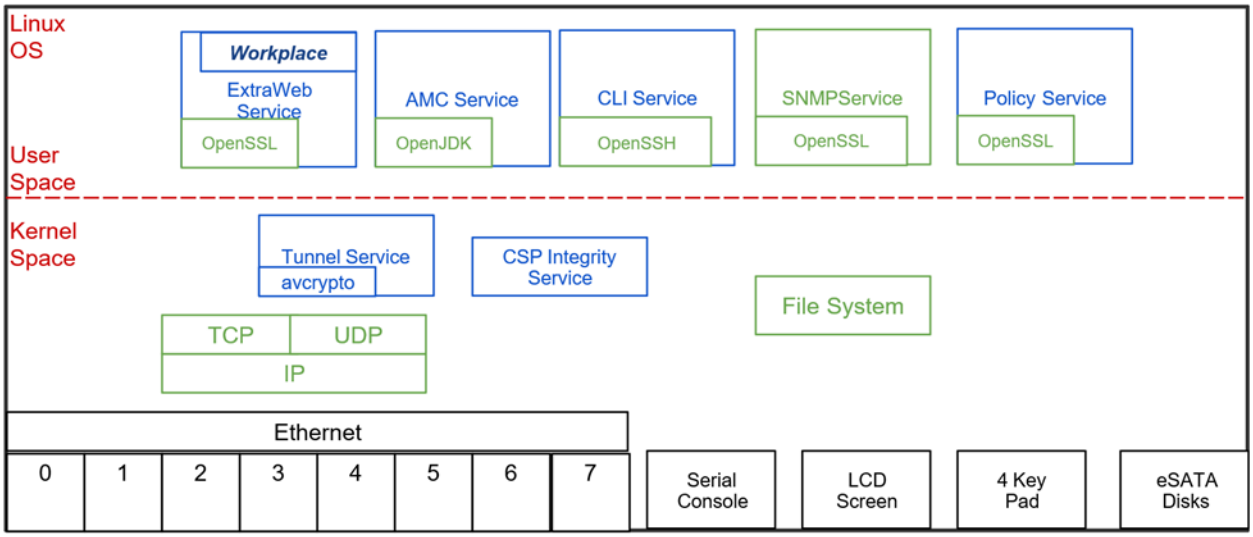

Figure 2: TOE Architecture

1.4.1.1 Operating System

The firmware includes the operating system which is based on the open source Linux 5.4 kernel. The Operating System provides a classic ring 3 protected user space for processes and a ring 0 privilege kernel space. The SMA firmware has components residing in both spaces.

1.4.1.2 File System

The File System component is a standard part of the Linux-based OS. This component provides persistent storage for all data including CSP's and is accessed through a standard API.

1.4.1.3 Networking Stack Component

The networking stack component includes support for Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) and it is a standard part of the Linux-based OS. This component provides the network transport services for all other components that require network connectivity and is accessed through a standard API.

1.4.1.4 SNMP Service Component

The SNMP Server component is standard part of the Linux-based OS. It provides read only SNMP monitoring services for the Cryptographic Module to external network monitoring systems. No CSPs are available for monitoring via the SNMP protocol .

1.4.1.5 CSP Integrity Service Component

This is a proprietary component developed by SonicWall. This component runs within the protected kernel space. It provides integrity checks for all CSP parameters including but not limited to Known Answer Tests, Continuous RNDRG validation, tamper detection and prevention and both Non-Fatal and Fatal FIPS Error state handling.

1.4.1.6 Tunnel and Avcrypto Service Component

This is a proprietary component developed by SonicWall. This component runs within the protected kernel space. It provides tunneling of traffic of all IP packets between End Users and resources located in the enterprise network. This component also implements cryptographic algorithms used by other components.

1.4.1.7 ExtraWeb and WorkPlace Service Component

This is proprietary component developed by SonicWall. This component runs as a standard multithreaded process in the user space. It provides access to web resources by proxying web requests between End Users and web resources located in the enterprise network. It utilizes OpenSSL for its transport communications and is compatible with industry standard browsers.

1.4.1.8 AMC Service Component

This is a proprietary component developed by SonicWall. This component runs as a Java application within the standard Java Runtime provided by the OpenJDK Java Runtime Environment (JRE) and runs as a standard process in the user space. it underpins remote administrative interface (AMC) and provides the services necessary for Security Administrator(s) to perform their respective roles associated with SF and CSPs. Cryptographic operations are provided by the OpenJDK JRE.

1.4.1.9 CLI Service Component

This is a proprietary component developed by SonicWall. It adds proprietary extensions to a set of open source CLI parsers provided by the Operating System. The CLI utilizes the Linux Bash Shell running as a standard process in the user space to execute each of the CLI commands.

1.4.1.10 Policy Service Component

This is a proprietary component developed by SonicWall. This component runs as a standard multithreaded process in the user space. It provides authentication and authorization services to all access methods.

1.4.2 *TOE Components*

1.4.2.1 Hardware

Table 2: SMA hardware appliances

Platform	Model	OS	CPU	RAM	Form	Specs
SMA v12.4.1	SMA 6210	SMA1000	Intel Core i5-7500 (Kaby Lake)	8GB (DDR4)	1U	6 1GB Ports
	SMA 7210	SMA1000	Intel Xeon E3-1275 v6 (Kaby Lake)	16GB (DDR4)	1U	6 1GB, 2 10GB SFP+ Ports

1.4.2.2 Software

The TOE, SonicWall SMA v12.4.1, is offered as SMA 6210 and SMA 7210 hardware appliances and SMA 8200v virtual appliance. The TOE's firmware is consistent across all appliances and

consists of multiple components, including SonicWall Operating System (SMA1000). See Figure 2: TOE Architecture for details. SonicWall Operating System, SMA1000, is based on Linux 5.4 kernel. The firmware assigned a uniquely identifiable build number and is the same for each appliance.

1.4.2.3 Virtualization

The TOE, SonicWall SMA v12.4.1, includes SMA 8200v virtual appliance. While SMA 8200v can be installed on a variety of hypervisors, it was only evaluated using the VMware ESXi 6.7 hypervisor running on a Dell PowerEdge R640, with the following virtual system specification:

Table 3: SMA virtual appliances

Platform	Model	Hypervisor	OS	CPU	RAM	Hard disk space	Virtual NIC
SMA v12.4.1	SMA 8200v	ESXi 6.7	SMA1000	4 vCPUs (Xeon Silver 4208 2.1GHz)	8GB ECC DDR-4 2400	160 GB, thick provisioned	2 vNIC of 1000BaseT

1.4.2.4 Management Interfaces

The TOE is configured and managed via a web-based Appliance Management Console (AMC) or a local Command Line Interface (CLI). The CLI is accessible from a directly-connected terminal while AMC is accessed remotely via web browser.

1.4.2.5 Physical Interfaces

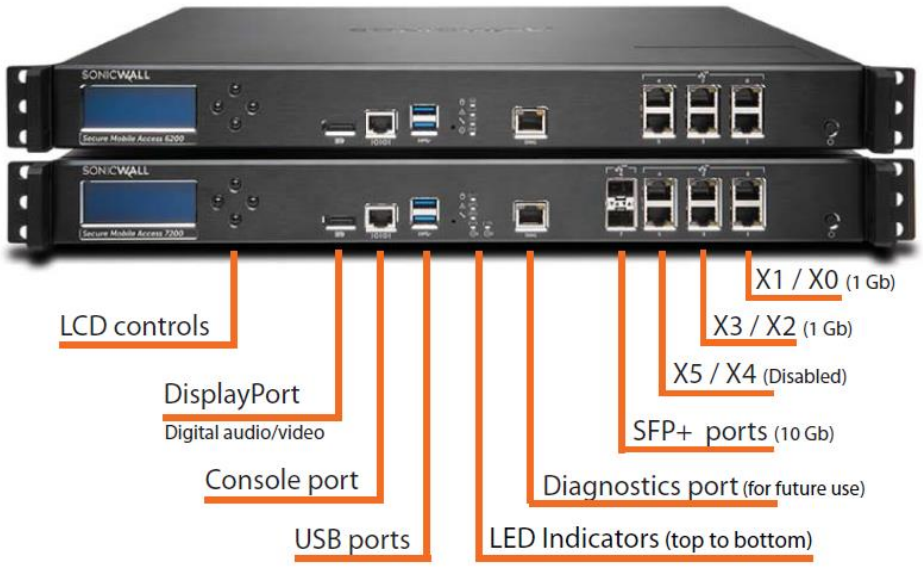

Figure 3: TOE Physical Interfaces

1.4.3 *Physical Boundary of the TOE*

The physical boundary of the hardware TOEs includes:

- The appliance hardware
  - RJ-45 to serial local management port (Console port)
  - USB port
  - Ethernet management port (X0 Ethernet port)

The physical boundary of the virtualised TOE is the virtual machine where the TOE is installed.

The Operational Environment of the TOE includes:

- The management workstation with a web browser
- VPN client (Connect Tunnel for Windows 10 v12.4.1)
- External IT servers:
  - Audit server for external storage of audit records
  - Certificate Authority and OCSP servers to support X.509 (optional)

1.4.4 *Deployment and Use*

The TOE can be deployed in either a single-homed or dual-homed configuration to fit the network architecture preference of the customer. In the single-homed configuration, a single network interface is used for both internal and external traffic. In such cases the appliance is usually installed in the demilitarized zone. In the dual-home configuration, one network interface is used for external traffic and the other interface is used for internal traffic. In either case, the TOE is deployed with access to both the WAN and to the LAN and corporate services for which remote access is desired, as well as to the IT entities in the operational environment such as syslog, etc.
Note: The SonicWall SMA does not provide firewall capabilities and should be secured behind a firewall

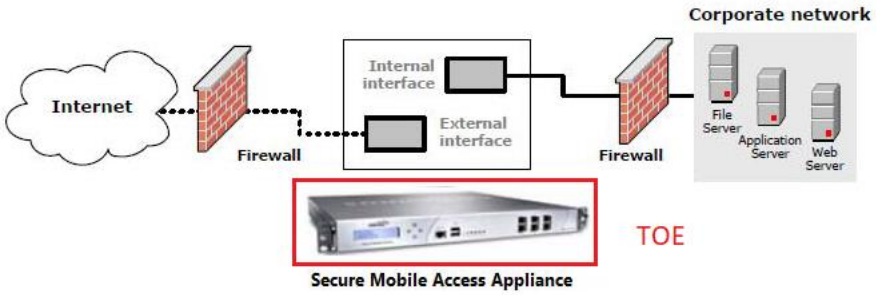

Figure 4**:** TOE Boundary and sample deployment

1.4.5 *Logical Boundary of the TOE*

The logical boundary of the TOE is defined by the implemented security functionality as summarized below. The TOE Security Functionality is further described in Section 7: TOE Summary Specification of this document. The SF is defined by the Security Functional Requirements (SFRs) listed in Section 6 of this document.

1.4.5.1 Security Audit

The TOE generates audit records for all security-relevant events. For each event, the TOE records the date and time, the type of event, the subject identity, and the outcome of the recoded event . The resulting records can be stored locally or securely sent to a designated audit server for archiving. Security Administrators using the appropriate AMC menu can also view audit records locally. The TOE also implements timestamps based on a local system clock to ensure reliable audit information produced.

1.4.5.2 Cryptographic Support

The TOE performs the following cryptographic functionality:

- Encryption, decryption, hashing, keyed-hash message authentication, random number generation, signature generation and verification utilizing dedicated cryptographic library
- Cryptographic functionality is utilized to implement secure channels
    - TLSv1.2
- Entropy is collected from multiple entropy sources and used to support PRNG seeding with full entropy
- Critical Security Parameters (CSPs) internally stored and cleared when no longer in use
- X.509v3 certificate-based authentication integrated with TLS protocol

The TOE is certified as a FIPS 140-2 level 2 cryptographic module, it internally manages CSPs and implements deletion procedures to mitigate the possibility of disclosure or modification of CSPs. Additionally, the TOE provides functionality to manually clear CSPs (e.g. host RSA keys), that can be invoked by a Security Administrator with appropriate permissions.

1.4.5.3 Identification and Authentication

The TOE implements 4 administrator roles managed by an AAA module that stores and manages permissions of all users and their roles. Before any other action, each user is identified with a login name and authenticated with a password. Each authorized user is associated with assigned role and specific permissions that determine access to TOE features.

1.4.5.4 Security Management

The TOE allows remote administration using a TLS session over an internal management Ethernet port and local administration using a console adapter via a separate RJ-45 running RS-232 signaling. Remote administration is conducted over web-based interface (AMC) and local administration conducted over CLI.

All of the management functionality is restricted to the Security Administrators of the TOE. The Security Administrators are authorized to perform configuration and management of the TOE. The term “Security Administrator” is used to refer to any user with an administrative role and sufficient permissions.

1.4.5.5 Protection of the TSF

The TOE implements a number of measures to protect the integrity of its security features. The TOE protects CSPs, including stored passwords and cryptographic keys, so they are not directly viewable in plaintext. The TOE also ensures that reliable time information is available for both log accountability and synchronization with the operating environment.

The TOE employs both dedicated communication channels as well as cryptographic means to protect the communication between itself and the other components in the operational environment.

The TOE performs self-tests to detect internal failures and to protect itself from malicious updates.

1.4.5.6 TOE Access

The TOE will display a customizable banner when an administrator initiates an interactive local or remote session. The TOE also enforces an administrator-defined inactivity timeout after which the inactive session is automatically terminated. Once a session (local or remote) has been terminated, the TOE requires the user to re-authenticate.

1.4.5.7 Trusted Path/Channels

The TOE protects remote sessions by establishing a trusted path secured with TLS between itself and the administrator. The TOE prevents disclosure or modification of audit records by establishing a trusted channel secured with TLS between itself and the audit server.

1.4.6 *Excluded Functionality*

The TOE supports a number of features that are not part of the evaluated functionality. These features are not included in the scope of the evaluation. The excluded TOE's functionality are listed as the following:

- The integration with a domain controller was not evaluated
- Any integration and/or communication with a single sign-on (SSO) provider was excluded from the evaluated configuration.
- The use of the SNMP management functionality is excluded and it is disabled by default. The use of SNMPv3 for monitoring is not restricted; however, it is not evaluated.
- The use of SMTP is not evaluated and should not be configured in the evaluated configuration.
- The remote access to CLI over SSH is not evaluated and not enabled in the evaluated configuration.
- The remote access to CLI via hypervisor console emulation is not evaluated, this configuration and mode of access is controlled by hypervisor software.
- The synchronization with an NTP server is not evaluated.
- The extraWeb and the WorkPlace interfaces and all relevant end-user functionality is not evaluated.
    - The interoperability with additional VPN clients, other than Connect Tunnel on Windows, is not evaluated
    - The access Policy setting and enforcement is not evaluated
    - The File Share functionality is not evaluated
    - The OnDemand Tunnel Agent is not evaluated
    - The Mobile Connect App integration is not evaluated
    - The Web Proxy Agent is not evaluated
- Limited controls via physical buttons on hardware appliance were not evaluated.
- The separation of security domains within SMA appliance was not evaluated, single-domain mode was configured and utilized throughout testing.

- The TOE was tested in a single-homed configuration, dual-homed configuration was not evaluated.
- The support for TLS 1.3 was not evaluated as corresponding SF and AAs are still being developed by NDcPP iTC and are not available in the current version of the cPP.
- The support for hypervisors other than ESXi was not evaluated.

1.4.7 *TOE Guidance and Reference Documents*

The following user guidance documents are provided to customers and are considered part of the TOE:

Table 4: TOE Reference Documents

Reference Title	ID
SonicWall Secure Mobile Access 12.4 Administration Guide	[ADMIN]
SonicWall SMA v12.4, Common Criteria Configuration Guide, Version 1.1 July 2021	[CC Addendum]

The documents in the following table were used as reference materials to develop this ST.

Table 5: ST Reference Documents

Reference Title	ID
Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002	[CC]
collaborative Protection Profile for Network Devices, Version 2.2e, 23 March 2020	[NDcPP]

2 Conformance Claims

2.1 Common Criteria Conformance Claim

This Security Target [ST] and the Target of Evaluation [TOE] are conformant to the following Common Criteria [CC] specifications:

- *Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components,* Version 3.1, Revision 5, April 2017, CCMB-2017-04-002
  - Part 2 Conformant with additional extended functional components as specified by the protection profile.

- *Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Components,* Version 3.1, Revision 5, April 2017, CCMB-2017-04-003
  - Part 3 Conformant with additional assurance activities as specified by the protection profile.

2.2 Protection Profile Claim

The TOE claims *exact* compliance to *collaborative Protection Profile for Network Devices, Version 2.2e, March 2020* [NDcPP].

2.2.1 *Technical Decisions*

- TD0592: NIT Technical Decision for Local Storage of Audit Records
  - Clarification of local audit storage requirements in distributed TOEs
  - Not applicable, the TOE is not a distributed appliance
- TD0591: NIT Technical Decision for Virtual TOEs and hypervisors
  - Clarification on virtual TOEs and hypervisors
  - Applicable
- TD0581: NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3
  - Additional selection to claim NIST SP 800-56Arev3
  - Applicable
- TD0580: NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e
  - Clarification about use of safe-prime DH groups in key establishment
  - Not applicable, only EC key establishment is claimed
- TD0572: NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers
  - Requirement to support dnsName identifiers in X509 certificates
  - Applicable
- TD0571: NiT Technical Decision for Guidance on how to handle FIA_AFL.1
  - Requirement to support password authentication with remote administration
  - Applicable
- TD0570: NiT Technical Decision for Clarification about FIA_AFL.1
  - Requirement to support password authentication with remote administration
  - Applicable
- TD0569: NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7
  - Instructions for handling multiple contexts for session resumption

    - Applicable
- TD0564: NIT Technical Decision for Vulnerability Analysis Search Criteria
    - Clarification on vulnerability search approach
    - Applicable
- TD0563: NIT Technical Decision for Clarification of audit date information
    - Clarification on date and time in audit records as part of FAU_GEN.1.2
    - Applicable
- TD0556: NIT Technical Decision for RFC 5077 question
    - Updated FCS_TLSS_EXT.1.4 test case 3(a)
    - Applicable
- TD0555: NIT Technical Decision for RFC Reference incorrect in TLSS Test
    - Discussion on whether Figure 3 in RFC 5077 is normative
    - Applicable
- TD0547: NIT Technical Decision for Clarification on developer disclosure of AVA_VAN
    - Additional developer disclosure of hardware and software components
    - Applicable
- TD0546: NIT Technical Decision for DTLS – clarification of Application Note 63
    - Clarification of Application Note 63
    - Not applicable, DTLS not claimed
- TD0538: NIT Technical Decision for Outdated link to allowed-with list
    - Updated 'allowed-with list' of PP-Modules
    - Not applicable, no modules claimed
- TD0537: NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3
    - Clarification of Application Note 113
    - Applicable
- TD0536: NIT Technical Decision for Update Verification Inconsistency
    - Additional documentation of each applicable update verification method
    - Applicable
- TD0528: NIT Technical Decision for Missing Eas for FCS_NTP_EXT.1.4
    - Additional NTP tests
    - Not applicable, NTP not claimed
- TD0527: Updates to Certificate Revocation Testing (FIA_X509_EXT.1)
    - Additional tests of X.509 certificates with ecPublicKey and explicit ecParameters
    - Applicable

2.3 Package Claim

The TOE does not claim to be conformant with any pre-defined packages.

2.4 Conformance Rationale

This Security Target claims strict conformance to only one PP – the NDcPP and no extended packages.

The Security Problem Definition (SPD) of this ST is consistent with the statement of the SPD in the PP, as the ST claims *exact* conformance to the PP and no other threats, organizational security policies, or assumptions are added.

The security objectives of this ST are consistent with the statement of the security objectives in the PP as the ST claims *exact* conformance to the PP and no other security objectives are added.

The security requirements of this ST are consistent with the statement of the security requirements in the PP as the ST claims *exact* conformance to the PP.

3 Security Problem Definition

3.1 Threats

This section identifies the threats applicable to the TOE as specified in the PP.

Table 6: TOE Threats

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

Threat Name	Threat Definition
Valid Updates	
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.
Audited Activity	
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.
Administrator and Device Credentials and Data	
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.
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.
Device Failure	
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 identifies assumptions applicable to the TOE as specified in the PP.

Table 7: TOE Assumptions

Assumption Name	Assumption Definition
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 and/or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP will not include any requirements on physical tamper protection or other physical attack mitigations. The cPP will not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/ services that could be deemed as general purpose computing. For example the device should not provide computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.
A.NO_THRU_TRAFFIC_PROTECTION	A standard/generic network device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the network device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the network device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs for particular types of Network Devices (e.g., firewall).

Assumption Name	Assumption Definition
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 being 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).
A.REGULAR_UPDATES	The network device firmware and software is assumed to be updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_SECURE	The administrator's credentials (private key) used to access the network device are protected by the platform on which they reside.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.
A.VS_TRUSTED_ADMINISTRATOR	The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.
A.VS_REGULAR_UPDATES	The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.VS_ISOLATION	For vNDs, it is assumed that VS provides, and is configured to provide sufficient isolation between software running in VMs one the same physical platform. Furthermore, it is assumed that the VS adequately protects itself form software running inside VMs on the same physical platform.
A.VS_CORRECT_CONFIGURATION	For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.

3.3 Organizational Security Policies

This section identifies the organizational security policies applicable to the TOE as specified in the PP.

Table 8: Organizational Security Policies

Policy Name	Policy Definition
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 defines the security objectives of the TOE and its supporting environment. The security objectives identify the responsibilities of the TOE and its supporting environment in meeting the security needs.

4.1 Security Objectives for the TOE

The collaborative Protection Profile for Network Devices v2.2edoes not define any security objectives for the TOE.

4.2 Security Objectives for the Operational Environment

This section identifies the security objectives as applicable to the operational environment as specified in the PP. These objectives

Table 9: Security Objectives for the Operational Environment

Objective Name	Environmental Security Objective Definition
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIAL_SECURE	The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

Objective Name	Environmental Security Objective Definition
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.
OE.VM_CONFIGURATION	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 Definition

The extended components listed in the Table 10 have been sourced from *collaborative Protection Profile for Network Devices, Version 2.2e, March 2020* [NDcPP].

The extended components, as defined in Section 8.3 of *Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model, Version 3.1, Revision 5*, are identified by "_EXT" in the component name. NDcPP Appendix C contains the definitions for all extended components.

5.1 Extended Security Functional Components

Table 10: Extended Components

Functional Component		
1	FAU_STG_EXT.1	Protected Audit Event Storage
2	FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
3	FCS_TLSS_EXT.1	TLS Server Protocol
4	FCS_TLSS_EXT.2	TLS Server support for mutual authentication
5	FCS_TLSC_EXT.1	TLS Client Protocol Without Mutual Authentication
6	FIA_PMG_EXT.1	Password Management
7	FIA_UIA_EXT.1	User Identification and Authentication
8	FIA_UAU_EXT.2	Password-based Authentication Mechanism
9	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
10	FIA_X509_EXT.2	X.509 Certificate Authentication
11	FIA_X509_EXT.3	X.509 Certificate Requests
12	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
13	FPT_APW_EXT.1	Protection of Administrator Passwords
14	FPT_TST_EXT.1	TSF Testing
15	FPT_STM_EXT.1	Reliable Time Stamps
16	FPT_TUD_EXT.1	Trusted Update
17	FTA_SSL_EXT.1	TSF-initiated Session Locking

5.2 Extended Security Functional Components Rationale

All extended security functional components are sourced directly from the NDcPP and applied verbatim. Exact compliance required by the NDcPP also mandates inclusion of all applicable extended components defined in the PP.

6 Security Requirements

6.1 Security Functional Requirements

Conventions
The following conventions have been applied in this document:

- **Security Functional Requirements** – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
  - **Iteration**: indicated by adding a string starting with "/" ("FCS_COP.1/Hash").
  - **Assignment**: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., **[assignment])**.
  - **Selection**: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., ***[selection]***).
  - **Refinement**: are identified with "**Refinement:**" right after the short name. Additions to the CC text are specified in ***italicized bold and underlined text***.

Note 1: Operations already performed in the cPP are not identified in this Security Target.

Note 2: Refinements made by the cPP authors will not be identified as refinements in this ST. The "Refinement" identifier is reserved for identifying any refinements made by the ST author.

- **Explicitly stated Security Functional Requirements** (i.e., those not found in Part 2 of the CC) are identified "_EXT" in the component name.)

The TOE security functional requirements are listed in Table 11. All SFRs are based on requirements defined in Part 2 of the Common Criteria or defined in the *collaborative Protection Profile for Network Devices, Version 2.2e, March 2020* [NDcPP].

Table 11: TOE Security Functional Components

Functional Components		
1	FAU_GEN.1	Audit Data Generation
2	FAU_GEN.2	User Identity Association
3	FAU_STG_EXT.1	Protected Audit Event Storage
4	FCS_CKM.1	Cryptographic Key Generation
5	FCS_CKM.2	Cryptographic Key Establishment
6	FCS_CKM.4	Cryptographic Key Destruction
7	FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
8	FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
9	FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
10	FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
11	FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
12	FCS_TLSS_EXT.1	TLS Server Protocol Without Mutual Authentication
13	FCS_TLSS_EXT.2	TLS Server support for mutual authentication
14	FCS_TLSC_EXT.1	TLS Client Protocol Without Mutual Authentication
15	FIA_PMG_EXT.1	Password Management

Functional Components		
16	FIA_AFL.1	Authentication Failure Management
17	FIA_UIA_EXT.1	User Identification and Authentication
18	FIA_UAU_EXT.2	Password-based Authentication Mechanism
19	FIA_UAU.7	Protected Authentication Feedback
20	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
21	FIA_X509_EXT.2	X.509 Certificate Authentication
22	FIA_X509_EXT.3	X.509 Certificate Requests
23	FMT_MOF.1/ManualUpdate	Management of Security Functions Behaviour
24	FMT_MTD.1/CoreData	Management of TSF data
25	FMT_MTD.1/CryptoKeys	Management of TSF data
26	FMT_SMF.1	Specification of Management Functions
27	FMT_SMR.2	Restrictions on Security Roles
28	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
29	FPT_APW_EXT.1	Protection of Administrator Passwords
30	FPT_TST_EXT.1	TSF Testing
31	FPT_TUD_EXT.1	Trusted Update
32	FPT_STM_EXT.1	Reliable Time Stamps
33	FTA_SSL_EXT.1	TSF-initiated Session Locking
34	FTA_SSL.3	TSF-initiated Termination
35	FTA_SSL.4	User-initiated Termination
36	FTA_TAB.1	Default TOE Access Banners
37	FTP_ITC.1	Inter-TSF Trusted Channel
38	FTP_TRP.1/Admin	Trusted Path

6.1.1 *Security Audit (FAU)*

FAU_GEN.1 Audit Data Generation

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

a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions comprising:
   - Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).
   - Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
   - Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
   - Resetting passwords (name of related user account shall be logged).
   - ***[no other actions]***
d) Specifically defined auditable events listed in ***Table 12***.

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of ***Table 12***.

Table 12: Auditable Events (Table 2 of the NDcPP)

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.
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FCS_TLSS_EXT.1	Failure to establish a TLS session	Reason for failure
FCS_TLSS_EXT.2	Failure to authenticate the client	Reason for failure
FCS_TLSC_EXT.1	Failure to establish a TLS Session	Reason for failure
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, 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/Rev	Unsuccessful attempt to validate a certificate	Reason for failure of certificate validation
	Any addition, replacement or removal of trust anchors in the TOE's trust store	Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.

FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None.
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process.	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FTA_SSL_EXT.1 (if “terminate the session” is selected)	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	Initiation of the trusted channel.	Identification of the initiator and target of failed trusted channels establishment attempt.
	Termination of the trusted channel.	
	Failure of the trusted channel functions.	
FTP_TRP.1/Admin	Initiation of the trusted path.	None.
	Termination of the trusted path.	
	Failures of the trusted path functions.	

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 ***[*****[delete all log files older than 7 days]*****]*** when the local storage space for audit data is full.

6.1.2 *Cryptographic Support (FCS)*

FCS_CKM.1 Cryptographic Key Generation (for asymmetric keys)

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

[

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

]

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

FCS_CKM.2 Cryptographic Key Establishment

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

[

- ***Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";***

]

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

FCS_CKM.4 Cryptographic Key Destruction

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

- For plaintext keys in volatile storage, the destruction shall be executed by a ***[single overwrite consisting of [zeroes]];***
- For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that
  - ***[logically addresses the storage location of the key and performs a [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, GCM]*** mode and cryptographic key sizes ***[128 bits, 256 bits]*** that meet the following: AES as specified in ISO 18033-3, ***[CBC as specified in ISO 10116, GCM as specified in ISO 19772]***.

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

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 (rDSA) with a key size (modulus) [2048 bits, 3072 bits]***
- ***Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits or greater]***

] that meets the following: ***[***

- ***For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3***
- ***For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384]' 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]*** ~~and cryptographic key sizes [assignment: cryptographic key sizes]~~ and **message digest sizes *[160, 256, 384]* bits** that meet the following: *ISO/IEC 10118-3:2004.*

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

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]*** and cryptographic key sizes **[160, 256]** and **message digest sizes *[160, 256]* bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC 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 ***[CTR_DRBG (AES)]***.

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from ***[*****[one]*** ***software-based noise source]*** with a minimum of ***[256 bits]*** of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

FCS_TLSC_EXT.1 TLS Client Protocol Without Mutual Authentication

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

[

- ***TLS_ECDHE_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_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289***
- ***TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289***

].

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifiers of the following types: ***[the reference identifier per RFC 6125 section 6, IPv4 address in CN or SAN]*** are matched to reference identifiers.

FCS_TLSC_EXT.1.3 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also:

[

- ***Not implement any administrator override mechanism***

].

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

FCS_TLSS_EXT.1 TLS Server Protocol Without Mutual Authentication

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

[

- ***TLS_ECDHE_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_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289***
- ***TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289***

].

FCS_TLSS_EXT.1.2 The TSF shall deny connections from client requesting SSL 2.0, SSL 3.0, TLS 1.0 and ***[TLS 1.1]***.

FCS_TLSS_EXT.1.3 The TSF shall ***[ECDHE curves [secp256r1, secp384r1] and no other curves]***.

FCS_TLSS_EXT.1.4 The TSF shall support ***[no session resumption or session tickets, session resumption based on session tickets according to RFC 5077]***.

FCS_TLSS_EXT.2 TLS Server Protocol for Mutual Authentication

FCS_TLSS_EXT.2.1 The TSF shall support TLS communication with mutual authentication of TLS clients using X.509v3 certificates.

FCS_TLSC_EXT.2.2 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the client certificate is invalid. The TSF shall also:

[

- ***Not implement any administrator override mechanism***

].

FCS_TLSS_EXT.2.3 The TSF shall not establish a trusted channel if the identifier contained in a certificate does not match an expected identifier for the client. If the identifier is a Fully Qualified Domain Name (FQDN), then the TSF shall match the identifier according to RFC 6125, otherwise the TSF shall parse the identifier from the certificate and match the identifier against the expected identifier of the client as described in the TSS.

6.1.3 *Identification and Authentication (FIA)*

FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within **[1-127]** 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 remote session using any authentication method that involves a password 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* **[256]** *characters.*

FIA_UIA_EXT.1 User Identification and Authentication

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

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

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

FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local ***[password-based]*** 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 certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using ***[the Online Certificate Status Protocol (OCSP) as specified in RFC 6960].***
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
    - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
    - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
    - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

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

FIA_X509_EXT.2 X.509 Certificate Authentication

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

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a

certificate, the TSF shall ***[not accept the certificate]***.

FIA_X509_EXT.3 X.509 Certificate Requests

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

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

6.1.4 *Security Management (FMT)*

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

FMT_MTD.1/CryptoKeys Management of TSF Data

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to manage the cryptographic keys to Security Administrators.

FMT_SMF.1 Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using* ***[digital signature, hash comparison]*** *capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- ***[***
  - ***Ability to modify the behavior of the transmission of audit data to an external IT entity;***
  - ***Ability to manage the cryptographic keys;***
  - ***Ability to configure the cryptographic functionality;***
  - ***Ability to set the time which is used for time-stamps;***
  - ***Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;***
  - ***Ability to import X.509v3 certificates to the TOE's trust store;***

  ]

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.

6.1.5 *Protection of the TSF (FPT)*

FPT_SKP_EXT.1 Protection of TSF Data (for reading of all symmetric keys)

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

FPT_APW_EXT.1 Protection of Administrator Passwords

FPT_APW_EXT.1.1 The TSF shall store administrative passwords in non-plaintext form.
FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext administrative passwords.

FPT_TST_EXT.1 TSF Testing

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests ***[during initial start-up (on power on), at the conditions*** **[as specified by FIPS PUB 140-2 Section 4.9.2]***]* to demonstrate the correct operation of the TSF**: [**
Power-up self-tests:
Integrity check of the cryptographic module
Known Answer Tests (KAT) of cryptographic primitives
Conditional self-tests:
Key generation pairwise consistency tests
Continuous random number generator testing
].

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 a means to verify firmware/software updates to the TOE using a ***[digital signature mechanism, published hash]*** prior to installing those updates.

FPT_STM_EXT.1 Reliable Time Stamps

FPT_STM_EXT.1.1 The TSF shall be able to provide reliable time stamps for its own use.
FPT_STM_EXT.1.2 The TSF shall ***[allow the Security Administrator to set the time].***

6.1.6 *TOE Access (FTA)*

FTA_SSL_EXT.1 TSF-initiated Session Locking

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

- ***terminate the session]***

after a Security Administrator-specified time period of inactivity.

FTA_SSL.3 TSF-initiated Termination

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

FTA_SSL.4 User-initiated Termination

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

FTA_TAB.1 Default TOE Access Banners

FTA_TAB.1.1 Before establishing an administrative user session the TSF shall display a Security Administrator-specified advisory notice and consent warning message regarding use of the TOE.

6.1.7 *Trusted Path/Channels (FTP)*

FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1 The TSF shall be capable of using ***[TLS]*** **to** provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, ***[VPN client]*** 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 **[transmitting audit records to an audit server, tunneling of traffic between VPN client and the enterprise network].**

FTP_TRP.1/Admin Trusted Path

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

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.

6.2 Security Assurance Requirements

Table 13: TOE Security Assurance Components

Assurance Components		
1	ASE_CCL.1	Conformance claims
2	ASE_ECD.1	Extended components definition
3	ASE_INT.1	ST introduction
4	ASE_OBJ.1	Security objectives for the operational environment
5	ASE_REQ.1	Stated security requirements
6	ASE_SPD.1	Security problem definition
7	ASE_TSS.1	TOE summary specification
8	ADV_FSP.1	Basic functional specification
9	AGD_OPE.1	Operational user guidance
10	AGD_PRE.1	Preparative procedures
11	ALC_CMC.1	Labeling of the TOE
12	ALC_CMS.1	TOE CM coverage
13	ATE_IND.1	Independent testing – conformance
14	AVA_VAN.1	Vulnerability survey

All applicable Security Assurance Requirements are specified in the cPP Section 7.

7 TOE Summary Specification

This chapter describes the security functions:

Table 14: TOE Security Functions

Security Objectives	SFR
7.1 Security Audit	FAU_GEN.1
	FAU_GEN.2
	FAU_STG_EXT.1
7.2 Cryptography	FCS_CKM.1
	FCS_CKM.2
	FCS_CKM.4
	FCS_COP.1/*
	FCS_RBG_EXT.1
	FCS_TLSS_EXT.1
	FCS_TLSS_EXT.2
	FCS_TLSC_EXT.1
7.3 Identification and Authentication	FIA_AFL.1
	FIA_PMG_EXT.1
	FIA_UIA_EXT.1
	FIA_UAU_EXT.2
	FIA_UAU.7
	FIA_X509_EXT.1/Rev
	FIA_X509_EXT.2
	FIA_X509_EXT.3
7.4 Security Management	FMT_MOF.1/ManualUpdate
	FMT_MTD.1/CoreData
	FMT_MTD.1/CryptoKeys
	FMT_SMF.1
	FMT_SMR.2
7.5 Protection of the security functionality	FPT_SKP_EXT.1
	FPT_APW_EXT.1
	FPT_TST_EXT.1
	FPT_TUD_EXT.1
	FPT_STM_EXT.1
7.6 TOE access	FTA_SSL_EXT.1
	FTA_SSL.3
	FTA_SSL.4
	FTA_TAB.1
	FTP_ITC.1

Security Objectives	SFR
7.7 Trusted path/channels	FTP_TRP.1/Admin

7.1 Security Audit

FAU_GEN.1, FAU_GEN.2, FAU_STG_EXT.1

The TOE is a standalone network device that generates syslog-conformant audit records whenever a management function, as defined in FMT_SMF.1, or auditable event, as defined in FAU_GEN.1, occurs.

The TOE supports six levels of events: Fatal, Error, Warning, Info (the default setting), Verbose, and Debug. The Security Administrator must ensure the level of the audit logging is set to Info to generate all audit records specified in the CC evaluated configuration. For each audited event, the date and time, the type of event, the subject identity (e.g. IP address or user identity), and the outcome are logged. The audit records may also contain event-specific content.

The TOE generates audit records for the following administrative tasks related to cryptographic keys:

- The generation and the destruction of a public and associated private key used to authenticate TOE's TLS server. Multiple key pairs can exist and identified in the audit records by CN.
- The installation and the removal of a trusted root or intermediate authority certificate(s) are identified in the audit records by CN.
- The generation of CSR and the import of a signed certificate used to authenticate TOE's TLS client. These are unique and identified in the audit records by CN.

All audit events recorded locally on the appliance and can also be duplicated over secure channel to an external audit server. On-device audit records reside on a separate 128G /var/log partition. Each log (i.e. management actions, policy, access_servers) exists as a set of 168 log files that collectively operate as a circular archive. Log files are rotated into archive daily with the oldest file overwritten first. In case the /var/log partition becomes full, all archived log files older than 7 days are deleted.

The viewing and the clearing of the local audit trail is restricted to Security Administrators with appropriate permissions in the Monitoring Information category. Manually clearing local audit trail wipes all audit records. In this way, the audit records are protected against unauthorized access and deletion.

The TOE is designed to securely forward audit records to a designated external audit server over a persistent trusted channel. This external audit server is authenticated by checking its X.509v3 certificate and secured with TLS protocol.

When configured, the TOE uploads audit records in syslog (RFC 5424) format, as they are generated, without any delay. If the connection to the external audit server is lost, the TOE will continue to save the local audit logs so there is no loss of audit. However, when the connection to the audit server is restored, the TOE only forwards the newly generated audit records. There is no automated log reconciliation process (syncing) between the locally stored records with the external audit server upon the re-establishment of the connection.

7.2 Cryptography

FCS_CKM.1, FCS_CKM.2, FCS_CKM.4, FCS_COP.1 (1-4), FCS_RBG_EXT.1, FCS_TLSC_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2

The TOE utilizes cryptographic primitives algorithms validated according to the NIST Cryptographic Algorithm Validation Program, ACVP certificates A1338, A1352, and A1358. These algorithm certificates identify SMA1000 operating system and CPUs included in SonicWALL SMA v12.4.1 SMA 6210, SMA 7210, and SMA 8200v appliances (See Table 2 and Table 3). These algorithm certificates covers RSA, EC, AES, SHA, HMAC, DRBG and TLS key establishment functionality and includes avcrypto (kernel) A1338, libcrypto (OpenSSL) A1352, and OpenJDK (Java) A1358 implementations.

The Avcrypto library is a kernel cryptographic module and is primarily used for entropy gathering and random number generation invoked by other modules. The OpenJDK, a Java-based cryptographic module, is used in securing remote web-based administration (AMC) with TLS Server. It is also responsible for secure key generation and certificate storage (via Java KeyStore). The Libcrypto (OpenSSL) is used to provide TLS Client functionality for secure communication with external IT servers. This module also provides X.509 certificate validation services to other components of the TOE.

The following Cryptographic Algorithm Validation Program (CAVP) certificates also applicable to the TOE:

Table 15: SonicWall SMA Cryptography

Requirement Class	Requirement Component	Implementation Details	Module	Certificate #
FCS: Cryptographic Support	FCS_CKM.1 Cryptographic Key Generation	Generating 2048-bit and 3072-bit RSA keypairs conforming to FIPS186-4. Generating P-256 and P-384 ECDSA keypairs conforming to FIPS 186-4.	OpenJDK libcrypto	A1358 A1352
	FCS_CKM.2 Cryptographic Key Establishment	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”;	OpenJDK libcrypto	A1358 A1352

Requirement Class	Requirement Component	Implementation Details	Module	Certificate #
		KAS-ECC-SSC Sp800-56Ar3 P-256, P-384		
	FCS_CKM.4 Cryptographic Key Destruction	Destruction of all keys is performed by single direct overwrite followed by a read-verify action.	All	N/A
	FCS_COP.1/Data Encryption Cryptographic Operation (AES Data Encryption/Decryption)	AES encryption and decryption used in CBC, GCM mode with 128-bit, 256-bit key sizes validated conforming to FIPS PUB 197.	Avcrypto OpenJDK libcrypto	A1338 A1358 A1352
	FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)	RSA signature generation and verification according to RSASSA-PKCS1v1_5 with 2048-bit and 3072-bit key sizes utilizing SHA-1 (protocol only), SHA-256. ECDSA signature generation and verification using P-256, P-384 curves and SHA-256, SHA-384 hash algorithms.	OpenJDK libcrypto	A1358 A1352
	FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)	Hashing using SHA-1, SHA-256, SHA-384 validated conforming to FIPS 180-4, Secure Hash Standard (SHS).	Avcrypto OpenJDK libcrypto	A1338 A1358 A1352
	FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)	Keyed hash HMAC-SHA1, HMAC-SHA256 validated conforming to FIPS 198, Keyed-Hash Message Authentication Code (HMAC).	Avcrypto OpenJDK libcrypto	A1338 A1358 A1352

Requirement Class	Requirement Component	Implementation Details	Module	Certificate #
		Supported cryptographic key sizes: 160, 256 bits and message digest sizes: 160, 256 bits. Keyed hash use matches validated hash algorithms implemented by the module.		
	FCS_RBG_EXT.1 Cryptographic Operation (Random Bit Generation)	CTR_DRBG (AES-256) random bit generation validated conforming to NIST SP PUB 800-90A.	avcrypto	A1338
	FCS_TLSC_EXT.1 FCS_TLSS_EXT.1 FCS_TLSS_EXT.2 TLS	KDF TLS TLSv1.2 with SHA-256	OpenJDK libcrypto	A1358 A1352

The TOE's cryptographic modules implement a number of approved and allowed algorithms but they are not utilized in the evaluated configuration. In the evaluated configuration, all cryptographic modules operate in the FIPS mode.

The TOE uses a software-based random bit generator (CTR_DRBG as implemented by avcrypto) that complies with NIST SP 800-90A for all cryptographic operations. Each DRBG instance is seeded with full entropy sourced from CPU Time Jitter Based Non-Physical TRNG (aka Jitter Entropy) with a minimum of 256-bits of entropy. The Jitter Entropy is compiled into a Linux Kernel Module that is loaded at boot time as part of DRBG initialization. This entropy source is used solely for the purpose of seeding and reseeding PRNGs (i.e. CTR_DRBG in FIPS mode). Jitter Entropy is a software-based mechanism that relies on the timing of unpredictable events. These events take the form of CPU execution jitter – measurable differences in the time it takes the CPU to execute a given set of instructions. Jitter Entropy is on-demand entropy source and does not accumulate entropy or preserve entropy across system reboots.

The TOE implements TLS that follows key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3 as part of ECDSA-based key establishment with both EC and RSA keys. In evaluated configuration, RSA keys are only used for authentication (i.e. ECDHE-based ciphers do not use RSA keys for key establishment).

In the evaluated configuration the TOE supports TLS v1.2 secure communication protocol that conforms to RFC 5246. The TOE will reject all connection attempts using TLS versions other than

TLS v1.2. The TOE acts as a TLS server during remote administration (AMC) and accepting VPN client (Connect Tunnel) connections.

When acting as a server, the TOE authenticates itself with a certificate chain that contains TOE's RSA-based or EC-based X.509v3 certificates. With VPN client, the TOE implements mutual authentication and requests client's certificate as part of the handshake. If connecting client fails to present a certificate, the connection is rejected with no fallback authentication option.

VPN client's identity certificate is first cryptographically validated, confirmed as issued by a trusted CA, checked for revocation, and then presented identifiers in the client's certificate comparted to configured identifiers for the specified AMC group. The client identifiers are configured via MGMT_CSFC_CERTIFICATE_REQUIRED_ATTRIBUTES CEM parameter where individual RDNs are separated by '&&' (e.g. 'C=US && O=SonicWall Inc.'). If any of the attributes do not match the configured values the client certificate will be rejected. Trusted CAs have to be imported into the TOE and manually added to the TOE's trust store. If the chain of trust cannot be established the client certificate will be rejected. Connection with VPN clients has multiple contexts – probe, API, and Tunnel. Probe is there to establish if mutual authentication is configured and will always result in a failed handshake in the evaluated configuration. The API context handles domain authentication and supports renegotiation. The Tunnel context is used for transport and is only possible after API context successfully established.

The TOE supports session resumption in API context based on session tickets according to RFC 5077. Session tickets adhere to the RFC 5077 ticket structure of NAME[16], IV[16], STATEDATA[varies], HMAC[32], where the hash algorithm is SHA256 and the cipher is AES-256-CBC.

The TOE acts as a TLS client when securely connecting to an audit server. In this case, the TOE validates presented server certificate prior to finishing TLS handshake. The following NIST curves are presented by the TOE as part of the Client Hello message – secp256r1 and secp384r1. The TOE supports 3 elliptic curves point formats – uncompressed (default), ansiX962_compressed_prime, ansiX962_compressed_char2.

When acting as a server, the TOE generates EC Diffie-Hellman parameters over NIST curves secp256r1 and secp384r1. The Server Key Exchange Message implements key agreement parameters according to RFC 5246 Section 7.4.3

The TOE supports the following ciphers with both TLS server and client:

- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

TLS_ECDHE_RSA ciphers used when RSA X.509 certificate is loaded, TLS_ECDHE_ECDSA ciphers used when EC X.509 certificate is loaded.

The TOE implements reference identifier matching according to RFC 6125 and also supports IPv4 addresses as identifiers. When connecting to an external entity, the reference identifier is specified during configuration of TLS connection. As part of negotiating TLS connection, the TOE will verify that server's certificate Subject Alternative Name (SAN) or Common Name (CN) contains expected identifier. Supported reference identifiers are FQDN or IPv4 addresses as defined in RFC 3986 in

CN or SAN. The CN is checked only if certificate does not contain SAN extension. When the TOE is presented with a certificate that does not have SAN and CN contains a single IP addresses presented, the TOE performs binary comparison between presented and reference identifiers. The TOE is also capable parsing DNS identifiers that include wildcards. The TOE only establishes connection if the peer certificate is valid, trusted, not revoked, and contains an identifier matching reference identifier.

The TOE does not implement certificate pinning and does not support Elliptic Curve Extension in the evaluated configuration.

The TOE is designed to destroy Critical Security Parameters (CSPs) when no longer required for use, to mitigate the possibility of disclosure. At various times, during TOE operation (e.g. an active TLS session) CSPs are present in RAM in plain text, then de-allocated and cleared from memory when no longer needed (e.g. on TLS session termination). Some CSPs (e.g. long term private keys) are also stored on disk and cleared when no longer used. The following table identifies applicable CSPs and summarizes zeroization procedure:

Table 16: SonicWall SMA CSPs

Identifier	Name	Generation / Algorithm	Purpose	Storage Location	Zeroization Summary
TLS-AMC-Priv	Private Key	PKCS1v1_5 / RSA or FIPS PUB 186-4 Appendix B.4/ECC	X509 private key used with certificate-based authentication	RAM (plain text) Disk (ciphertext)	Single direct overwrite consisting of zeros followed by a read-verify action.
TLS-SENC	TLS Session Keys	Generated using TLS KDF	Symmetric keys for TLS	RAM (plain text)	Cleared when device is powered down or as part of session termination. Overwritten by a new value.
AUTH-PW	Authentication Passwords	SHA256	Credentials used to authenticate the administrator login.	Disk (cipher text)	Hashed passwords exist in a local database and replaced when changed and saved. The passwords are stored in the ciphertext (hash and salt) form only. Overwritten by a new value

Identifier	Name	Generation / Algorithm	Purpose	Storage Location	Zeroization Summary
				RAM (cipher and plain text)	Passwords in RAM are zeroized when creating / resetting the password. Both clear text and encrypted forms are stored in RAM. Overwritten by new value.
DRBG-EI	PRNG Seed key	/dev/random	Seed key for PRNG	RAM (plain text)	Cleared when device is powered down or during reboot by the new seed.
OS-KEK	Keystore encryption key	Platform	Used to encrypt CSPs in certificate storage	RAM (plain text) Disk (plain text)	In RAM, cleared when device is powered down or during reboot. On disk, overwritten by zeroization.

TLS-AMC-Priv is a private key associated with TOE's X509 certificate. It exists in non-volatile memory as ciphertext and is part of encrypted java keystore that is in turn protected with OS-KEK key.

7.3 Identification and Authentication

FIA_PMG_EXT.1, FIA_AFL.1, FIA_UIA_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7, FIA_X509_EXT.1/Rev, FIA_X509_EXT.2, FIA_X509_EXT.3, FMT_MTD.1/CryptoKeys, FCS_TLSC_EXT.1, FCS_TLSS_EXT.1

The TOE functionality can be logically divided into following two categories: End User and Control and Configuration. The Control and Configuration functionality is associated with a dedicated physical interface (X0 Ethernet port as shown in Figure 3: TOE Physical Interfaces), is intended to be LAN-facing, and implements TOE's administrative functionality via AMC (Appliance Management Console). The AMC is an HTML-based administrative interface behind an authentication prompt.

The End User functionality is associated with a dedicated physical interface (X1 Ethernet port as shown in Figure 3: TOE Physical Interfaces), is intended to be WAN-facing, and implements end-user access gateway functionality via WorkPlace web portal. The WorkPlace is an HTML-based end-user interface behind an authentication prompt. This WorkPlace functionality does not include any management or configuration options and does not directly interface with the Control and Configuration.

The TOE requires any user to be identified and authenticated before any action. The warning banner is displayed before login prompt on any of the management access points (local CLI or remote AMC interfaces). In the evaluated configuration, the TOE does not allow unauthenticated configuration of

the TOE's network routing/switching services and does not allow any unauthenticated management actions.

The TOE permits an administrator to configure the number of unsuccessful authentication attempts within a range of 1 to 127 as well as time allowed before a retry is permitted from 1 to 1440 minutes during which, the authenticating user is locked out. The TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, by distinguishing between local and remote login attempts.

A requesting user will be prompted to enter a user name and password upon establishing successful connection. The TOE will then compare entered credentials against the local user database. If the combinations match, the TOE will then attribute (bind) the administratively assigned role (predetermined group of privileges that dictate access to TOE functions) to that user for the duration of the interactive session.

For a local administrative session, password character entries are not echoed to the screen. For a remote administrative session, credentials during transmission are protected by a secure channel. Passwords can be composed of any number of upper and lower case letters, numbers, and the following special characters: ["!", "@", "#", "$", "%", "^", "&", "*", "(", ")", ["_", "+","-", "=", "{", "}", "[", "]", "\", ":", ";", "<", ">", "?", "/"]].

Administrative (management) roles are created with specific job functions in mind. Through these roles, users acquire the permissions to perform their associated job function. If a user's role matches one of the allowed roles for a specific management functionality, then access to such functionality is granted. Multiple users can be assigned the same role, but each user can be assigned only a single role.

By default, TOE provides four system-defined administrator roles:

- Primary Admin - This administrator security role has a full access to all administrative functionality, can create and delete other administrator accounts, and can access CLI.
- Super Admin - This administrator security role has a full access to all administrative functionality and can create and delete other administrator accounts.
- Security Admin - This administrator security role has read/write access to security administration and monitoring pages in AMC, and view access to system settings.
- System Admin -This administrator security role has read/write access to system and monitoring pages, and view access to security pages.

For remote administration, implemented as a web-based interface secured with TLS, the TOE configured to authenticate itself with X509 certificates. For both local and remote administration, only username and password-based authentication is supported in the evaluated configuration.

Upon successful authentication, the TOE assigns administratively defined role to that user for the duration of the session. Successful login is indicated by TOE offering a home page or a command line prompt.

The TOE supports having a minimum of 15 character password length and supports the utilization of upper and lower case, numeric, and special character combinations for password construction.

The TOE supports the use of X.509v3 certificates, as defined by RFC 5280, to authenticate connections with authorized IT entities and to authenticate itself to remote administrators. When certificate-based authentication is used with remote administrators, the TOE presents its server certificate along with a certificate chain. When certificate-based authentication is used with external IT entities, the TOE validates the presented certificate, checks the chain of trust against the TOE's internal trust store, and performs certificate revocation check.

- Certificate validation includes path validation (checking CA certificates in the chain), certificate processing (validating the signature, checking `keyUsage`), and extension processing (checking `basicConstraints` and `extendedKeyUsage` extensions).

- Verifying the chain of trust includes validating each certificate in the chain, verifying that each CA certificate has `basicConstraints` flag set to `CA:TRUE`, verifying that the certificate path terminates with a valid CA certificate designated as a trust anchor.

- Revocation checking is implemented using OCSP and is performed on the intermediate CA and leaf certificates. Regardless of a full chain or leaf/identity certificates being presented to the TOE, revocation is performed on the full chain up to a trust anchor as long as the TOE has all necessary CA certificates to determine the trust. Otherwise, the certificate is rejected as untrusted at an early stage of the certificate validation process.

If any of these steps fail, the connection is terminated at the handshake stage.

When an X.509v3 certificate is presented during the TLS handshake, the TOE validates the presented certificate and the entire trust chain up to a trust anchor by performing revocation checks. The revocation checks are performed by sending an OCSP requests to an OCSP responders specified in the AIA extension and verifying the OCSP's signed response. If the connection to a specified OCSP responder cannot be established the TOE's would reject the certificate.

The list of OCSP responders is specified in a CA certificate, in the `authorityInfoAccess` extension.

The TOE requires an X.509v3 certificate to authenticate its management interface (AMC) and support secure TLS connections with remote administrators. The TOE internally generates a 2048 bits or higher RSA or 256 bits or higher EC key pairs and uses the public key to produce a Certificate Signing Request (CSR). The CSR is then exported and signed by a CA to generate the TOE's X.509v3 certificate. The type of certificate generated will determine which ciphers are used (i.e. RSA vs. ECDSA authentication). In addition to the TOE's X.509v3 certificate, the signing CA's certificate must also be installed on the TOE. The TOE's administrator does not have direct access to RSA/EC private keys.

The TOE is capable of authenticating with both CA-signed and self-signed certificates, but in the evaluated configuration the TOE's X.509v3 certificate must be signed by a trusted CA. The TOE will verify that server's certificate Subject Alternative Name (SAN) or Common Name (CN) contains expected identifier. Supported reference identifiers are FQDN or IPv4 addresses as defined in RFC 3986 in CN or SAN.

The TOE's certificate must contain the following parameters:

- `subjectAltName` with an IP or FQDN identifier
- `keyUsage` with `keyEncipherment` and `digitalSignature` bits set
- `extendedKeyUsage` with `serverAuthentication` bits set

Creation of the CSR, exporting it for the CA to sign, and installation of the signed X.509v3 certificate must be performed by a Security Administrator. The TOE performs revocation checking on the signed CSR and all intermediate CAs at the time of loading.

Instructions for generating the TOE's CSR, and installing both the TOE's X.509v3 certificate and the CA's X.509v3 certificate, are provided in the Administration Guide.

The TOE supports the following methods to load a Certificate Signing Request (CSR) signed by a trusted CA:

- Upload the certificate in PEM format as a file
- Manually copy-paste the base-64 encoded certificate

When a Security Administrator is adding the CA certificate to the trust store, the certificate in PEM format is downloaded and installed. Prior to adding a new CA to its existing list of trusted certificates, the TOE verifies the following:

- Digital signature check on the certificate
- The `basicConstraints` extension is present with the CA flag set to `TRUE`
- The `keyUsage` has the `keyCertSign` bit is set
- The certificate has not expired

The TOE also supports mutual authentication with connecting VPN clients. In the evaluated configuration when connecting client fails to present a certificate, the connection is rejected with no fallback authentication option.

VPN client's identity certificate is first cryptographically validated, confirmed as issued by a trusted CA, checked for revocation, and then the presented identifiers in the client's certificate comparted to configured identifiers for the specified AMC group. The VPN client identifiers are configured via MGMT_CSFC_CERTIFICATE_REQUIRED_ATTRIBUTES CEM parameter where individual RDNs are separated by '&&' (e.g. 'C=US && O=SonicWall Inc.'). If any of the attributes do not match the configured values the client certificate will be rejected.

All certificates are stored in a private, persistent location on the TOE. Only Security Administrators with sufficient privileges have an ability to access, generate, or load a certificate.

7.4 Security Management

FMT_MOF.1/Manual Update, FMT_MTD.1/Core Data, FMT_SMF.1, FMT_SMR.2

The TOE supports remote administration via a web-based interface, called Appliance Management Console (AMC), secured with TLS channel and consigned to a dedicated physical interface (X0 Ethernet port as shown in Figure 3: TOE Physical Interfaces). The TOE supports local administration, command-line interface (CLI), via a directly connected console adapter connected to a dedicated RJ-45 console port.

The TOE is designed to be primarily managed via web-based AMC interface that offers all management functions through a GUI. The CLI is a command-line interface restricted to a limited subset of management functionality aimed at initial configuration (Setup Tool) and system status monitoring. The CLI permits authorized administrators to set system time, verify audit logs, and restart appliance.

The term “Security Administrator” is used to refer to any administrator account, by default Super Admin, Security Admin, and System Admin roles. By default, AMC is configured with a primary administrator that has full access to all areas of AMC. Only the primary administrator can add, edit, or delete other administrator accounts and roles. Multiple individuals can be assigned Security Administrator role. Each Security Administrator can manage the TOE via AMC. The access to the CLI requires Super Admin role.

The TOE requires each user to be successfully authenticated before allowing any other action on behalf of that user. Both the remote management interface (AMC) and the local management interface (CLI) require each Security Administrator to enter their credentials (username and password) for authentication and identification before any TSF-mediated actions can be performed on behalf of the Security Administrator.

By default, the TOE implements 4 administrator roles. Access and authorization is controlled based on a user’s role. All of the management functions are restricted to the Security Administrators of the TOE.

Security Administrators with appropriate permission can perform the following management actions: configure the access banner, configure the session inactivity timer, update the TOE, configure the authentication failure policy, configure audit behavior, manage the cryptographic keys, set time, manage the TOE’s trust store and import X.509v3 certificates, restart and shut down the TOE, and review the audit records.

The administrator role and its permission determines the functions the user can perform, the primary administrator can grant additional administrative controls to other roles. For defining administrator roles, the features in AMC are grouped into four categories. For each category, it is possible to specify the permissions granted to a role. The four categories of administrator permissions: security administration, system configuration, system maintenance, system monitoring. The permission level for each category can be set to one of the following: modify, view, none.

It is understood that not all administrators will have sufficient permissions assigned to them to perform each administrative function discussed in this document. Specifically, the TOE restricts the ability to perform manual update to the roles with modify permission level in system maintenance category. Likewise, the TOE restricts the ability to manage the TOE’s trust store X.509v3 certificates associated with AMC to the roles with modify permission level in system configuration category.

Manual update functionality is restricted to the Security Administrators with appropriate permissions. The TOE supports both published hash and digital signature based update. To initiate a manual update, Security Administrator first has to download a binary file from the https://www.mysonicwall.com After downloading the binary file, Security Administrator installs (and verifies) the file by logging into the AMC web interface, selecting System Configuration -> Maintenance, clicking Update in the System Software updates area, browsing to select the file, then clicking on Install Update. The TOE will return an error if the verification of the signature on the binary file fails. Alternatively, Security Administrator can perform a manual published hash comparison.

7.5 Protection of the security functionality

FPT_SKP_EXT.1, FPT_APW_EXT.1, FPT_TST_EXT.1, FPT_STM_EXT.1, FPT_TUD_EXT.1,

The TOE is a standalone appliance designed to function independently, as a result, both security functionality and measures to protect security functionality are focused on self-protection.

The TOE employs both a dedicated communication channels (i.e. separate physical RJ-45 ports for management) as well as cryptographic means (i.e. encrypted secure channels) to protect remote administration.

The TOE protects critical security parameters (CSP) such as stored passwords and cryptographic keys so they are not directly accessible via normal administrative interfaces. Locally stored password information is obscured by use of hashing (SHA256). Additionally, when login-related configuration information is accessed through local TOE interfaces it is obfuscated by representing input with a series of asterisks.

The TOE is a hardware appliance that implements hardware-based real-time clock managed by an embedded OS, which also controls the exposure of administrative functions. This clock is used to produce reliable timestamps that are available for audit trail generation, synchronization with the operational environment, session inactivity checks, and certificate expiration validation.

The TOE performs diagnostic self-tests during start-up and generates audit records to document failure. Some low-level critical failure modes can prevent TOE start-up and as a result will not generate audit records. In such cases, TOE appliance would enter failure mode displaying error codes, typically displayed on the console. By default, the TOE in the evaluated configuration (FIPS mode) will reboot or to stop with errors displayed when non-critical errors are encountered. The cryptographic module performs self-tests during startup; the messages are displayed on the console and audit records generated for both successful and failed tests. Self-tests comply with the FIPS 140-2 requirements for self-testing. The module performs known-answer algorithm testing, integrity testing, and conditional self-testing. Failure of any of the FIPS mode tests during boot process will stop start-up process and prompt the user to reload. For all start-up tests, successful completion is indicated by reaching operational status. The diagnostic self-tests monitor the TOE against a set of anticipated faults, therefore outside of failure mode induced by failing self-tests, the TSF is assumed to operate correctly.

Upgrading the TOE is a multi-step process that must be performed by a Security Administrator. An authorized user must authenticate to the secure support website where the software downloads are available. The downloaded image must be then transferred to the appliance using an administrative interface. The binary file can be manually verified using the published hash comparison. Upon successful comparison, the administrator can then initiate the upgrade. The TOE also performs verification of the signature contained in the binary file and returns an error message if the verification fails. The version of the current system software and the product serial number are displayed at the bottom of the left-hand navigation bar on every page in the remote administrative interface (AMC).

7.6 TOE access

FTA_SSL.3, FTA_SSL.4, FTA_SSL_EXT.1, FTA_TAB.1

The TOE implements local administrative access via command line interface (CLI) and remote administrative access via web-based AMC interface. In both cases, the TOE will display a customizable banner when an administrator initiates an interactive session either locally or remotely.

The TOE is designed to lock accounts after a number of unsuccessful login attempts. The TOE's minimum lockout value must be configured to a non 0 value to enforce an administrator-defined inactivity timeout after which the inactive session is automatically terminated. Once a session (local or remote) has been terminated, the TOE requires the user to re-authenticate. The administrator can force termination of current session by issuing the logout command `exit` with CLI or by clicking log out with AMC.

The TOE permits an administrator to configure the number of unsuccessful authentication attempts within a range of 1 to 127 as well as time allowed before a retry is permitted from 1 to 1440 minutes during which, the authenticating user is locked out.

7.7 Trusted path/channels

FTP_ITC.1, FTP_TRP.1

The TOE protects remote management sessions by establishing a trusted path (secured with TLS) between itself and the administrator connected to a dedicated RJ-45 LAN management port. When a client attempts to connect, the TOE and the client will negotiate the preferred and mutually acceptable secure ciphersuite offered by the server to protect the session. If the cipher can not be selected, or the protocol version cannot be agreed on, the connection is dropped. After initial connection, protocol negotiation, and key exchange, a symmetrical encryption is used to transmit encrypted application data. After successful TLS handshake all traffic between the TOE and the external entity is encrypted using AES-CBC-128 or AES-CBC-256 symmetric encryption algorithm. Password-based authentication is encapsulated in this encrypted TLS channel. For certificate-based authentication, RSA or EC key pair associated with TOE's X.509v3 server certificate is generated by the TOE and the private key is protected by the TSF.

The TOE protects communications with the audit server by establishing a trusted channel between itself and the audit server. To implement this trusted channel, the TOE uses TLS v1.2 protocol with certificate-based authentication. For certificate-based authentication, the presented certificate (x.509v3) is first cryptographically validated, confirmed as issued by a trusted CA, checked for revocation, and then identifiers compared. Trusted CAs have to be imported into the TOE and manually added to the TOE's trust store.

The TOE protects communications with the Connect Tunnel VPN client by establishing a mutually authenticated secure channel between itself and the client. To implement this trusted channel, the TOE uses TLS v1.2 protocol with certificate-based mutual authentication. The TOE supports session resumption with VPN clients based on session tickets according to RFC 5077.

8 Acronyms and Terminology

8.1 Acronyms

The following table defines CC and Product specific acronyms used within this Security Target.

Table 17: Acronyms

Acronym	Definition
CA	Certificate Authority
CC	Common Criteria
CSP	Critical Security Parameter
DN	X.500 Distinguished Names
FIPS	Federal Information Processing Standard
IT	Information Technology
NIST	National Institute of Standards and Technology
OE	Operational Environment
OS	Operating System
OSP	Organizational Security Policy
PP	Protection Profile
RDN	Relative DN
RFC	Request for Comment
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Function **TSF** = TOE for pND or Case 1 vND according to section 1.2 **TSF** = TOE + VS for Case 2 vND (vND evaluated as a pND) according to section 1.2
TSS	TOE Summary Specification

8.2 Product Acronyms and Terminology

The following table defines the CC and Product-specific terminology used within this Security Target.

Table 18: Terminology

Terminology	Definition
AAA	Authentication, Authorization, and Accounting (AAA). A security architecture for distributing systems for controlling remote access to services.
RADIUS	Remote Authentication Dial-In User Service (RADIUS) protocol that includes authentication and authorization.
RSA	Ron **R**ivest, Adi **S**hamir, Leonard **A**dleman. Public-key cryptosystem algorithm.

Terminology	Definition
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
SSH	Secure Shell
TLS	Transport Layer Security