Sonicwall SonicOS/X v7.0.1 with VPN and IPS on TZ, NSa, NSsp, and NSv Appliances Security Target

Document Version: 1.2

2400 Research Blvd

Suite 395

Rockville, MD 20850

Revision History

Version	Date	Changes
Version 1.0	16 February, 2024	Initial Release
Version 1.1	12 November, 2024	Updates after internal reviews
Version 1.2	24 December, 2024	Updates to address NIAP comments

Contents

1 Introduction

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

1.1 Security Target and TOE Reference

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

Table 1 – TOE/ST Identification

Category	Identifier
ST Title	Sonicwall SonicOS/X v7.0.1 with VPN and IPS on TZ, NSa, NSsp, and NSv Appliances Security Target
ST Version	1.2
ST Date	24 December 2024
ST Author	Acumen Security, LLC.
TOE Identifier	Sonicwall SonicOS/X v7.0.1 with VPN and IPS on TZ, NSa, NSsp, and NSv Appliances
TOE Version	7.0.1
TOE Developer	Sonicwall, Inc.
Key Words	Firewall, Intrusion Prevention System, Virtual Private Network Gateway, Stateful Traffic Filter Firewall.

1.2 TOE Overview

The TOE is comprised of the SonicWall SonicOS/X v7.0.1 software running either on purpose built TZ, NSa, NSsp, series hardware appliance platforms and NSv virtual appliances running on purpose built ESXi hardware appliances.

The appliance next generation firewall capabilities include stateful packet inspection. Stateful packet inspection maintains the state of network connections, such as Transmission Control Protocol (TCP) streams and User Datagram Protocol (UDP) communication, traveling across the firewall. The firewall distinguishes between legitimate packets and illegitimate packets for the given network deployment. Only packets adhering to the administrator-configured access rules are permitted to pass through the firewall; all others are rejected.

The appliance capabilities include deep-packet inspection (DPI) used for intrusion prevention and detection. These services employ stream-based analysis wherein traffic traversing the product is parsed and interpreted so that its content might be matched against a set of signatures to determine the acceptability of the traffic. Only traffic adhering to the administrator-configured policies is permitted to pass through the TOE.

The appliances support Virtual Private Network (VPN) functionality, which provides a secure connection between the device and the audit server. The appliances support authentication and protect data from disclosure or modification during transfer.

The appliances are managed through a web based Graphical User Interface (GUI). All management activities may be performed through the web management GUI via a hierarchy of menu buttons. Administrators may configure policies and manage network traffic, users, and system logs. The appliances also have local console access where limited administrative functionality to configure the network, perform system updates, and view logs.

1.3 TOE Description

This section provides an overview of the TOE architecture, including physical boundaries, security functions, and relevant TOE documentation and references.

The TOE supports secure connectivity with several other IT environment devices as described below.

Table 2 –Environmental Components for TOE

Component	Required	Usage/Purpose Description
TOE Hardware	Yes	The Sonicwall TZ, NSa, and NSsp, physical hardware models
TOE Virtual Hardware	Yes	Virtual hardware provided by VMware vShpere ESXi 7.0 and ESXi 8.0 on Dell PowerEdge R640.
Management Workstation	Yes	This includes any IT Environment Management workstation
Audit Server	Yes	An audit server supporting the syslog protocol with an IPsec peer supporting IKEv2 and ESP in the cryptographic protocols defined in 5.2.2.6 of this document.
Management Console	Yes	Any computer that provides a supported browser may be used to access the GUI

1.3.1 Physical Boundaries

The TOE is a software and hardware TOE. It is a combination of TZ, NSa, NSsp, and NSv hardware/software appliance and the SonicOS/X 7.0.1 software. The following table lists all the instances of the TOE that operate in the evaluated configuration. All listed TOE instances offer the same core functionality but vary in number of processors, physical size, and supported connections.

The physical boundary of the TOE includes the Sonicwall TZ, NSa, NSsp hardware models shown in Table 3 and the virtual appliances and related hardware shown in Table 4 running Sonicwall SonicOS/X software identified in table 1. The virtual appliances are evaluated as virtual Network Devices (vND) which is case 1 of Section 1.2 of NDcPP v2.2e. The physical TOE is shipped to the customer via commercial courier. The virtual appliance's deployment packages can be downloaded from the https://www.mysonicwall.com site.

Table 3 –Physical Boundary Components for TOE hardware models

Appliance Series	Appliance Model	Operational Environment	Microarchitecture
TZ	TZ 670	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570W	Marvell CN9130	Quad Core Armv8 Cortex-A72

Appliance Series	Appliance Model	Operational Environment	Microarchitecture
	TZ 570P	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 470	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 470W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270W	Marvell 88F7040	Quad core Armv8 Cortex-A72
NSa	NSa 2700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 3700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 4700	Intel Xeon D-2123IT	Skylake
	NSa 5700	Intel Xeon D-2123IT	Skylake
	NSa 6700	Intel Xeon D-2123IT	Skylake
NSsp	NSsp 10700	Intel Xeon D-2166NT	Skylake
	NSsp 11700	Intel Xeon D-2166NT	Skylake
	NSsp 13700	Intel Xeon D-2187NT	Skylake

Table 4 –Physical Boundary Components for TOE Virtual Appliance

Appliance Series	Appliance Model	Operational Environment
NSv	NSv 270	ESXi 7.0 and 8.0 on Dell PowerEdge R640 (Running on Intel Xeon Silver 4208 (Cascade Lake))
	NSv 470	
	NSv 870	

1.3.2 Security Functions Provided by the TOE

The TOE provides the security functions required by the Collaborative Protection Profile for Network Devices, hereafter referred to as NDcPP v2.2e or NDcPP, collaborative Protection Profile Module for Stateful Traffic Filter Firewall, hereafter referred to as MOD_FW v1.4e or MOD_FW, PP-Module for VPN Gateways Version 1.3 hereafter referred to as MOD_VPNGW v1.3 or MOD_VPNGW, PP-Module for Intrusion Protection Systems (IPS) Version 1.0, hereafter referred to as MOD_IPS v1.0 or MOD_IPS.

1.3.2.1 Security Audit

The TOE generates audit records for administrative activity, security related configuration changes, cryptographic key changes and startup and shutdown of the audit functions. The audit events are associated with the administrator who performs them, if applicable. The audit records are transmitted over an IPsec VPN tunnel to an external audit server in the IT environment for storage.

1.3.2.2 Cryptographic Support

The TOE provides cryptographic functions (key generation, key establishment, key destruction, cryptographic operation) to secure remote administrative sessions over Hypertext Transfer Protocol

Secure (HTTPS)/Transport Layer Security (TLS), and to support Internet Protocol Security (IPsec) to provide VPN functionality and to protect the connection to the audit server.

1.3.2.3 Residual Data Protection

The TOE ensures that data cannot be recovered once deallocated.

1.3.2.4 Identification and Authentication

The TOE provides a password-based logon mechanism. This mechanism enforces minimum strength requirements and ensures that passwords are obscured when entered. The TOE also validates and authenticates X.509 certificates for all certificate use.

1.3.2.5 Security Management

The TOE provides management capabilities via a Web-based GUI, accessed over HTTPS. Management functions allow the administrators to configure and update the system, manage users and configure the Virtual Private Network (VPN) and Intrusion Prevention System (IPS) functionality.

1.3.2.6 Protection of the TSF

The TOE prevents the reading of plaintext passwords and keys. The TOE provides a reliable timestamp for its own use. To protect the integrity of its security functions, the TOE implements a suite of self-tests at startup and shuts down if a critical failure occurs. The TOE verifies the software image when it is loaded. The TOE ensures that updates to the TOE software can be verified using a digital signature.

1.3.2.7 TOE Access

The TOE monitors local and remote administrative sessions for inactivity and either locks or terminates the session when a threshold time period is reached. An advisory notice is displayed at the start of each session.

1.3.2.8 Trusted Path/Channels

The TSF provides IPsec VPN tunnels for trusted communication between itself and an audit server. The TOE implements HTTPS for protection of communications between itself and the Management Console.

1.3.2.9 Intrusion Prevention

The TOE performs analysis of IP-based network traffic and detects violations of administratively defined IPS policies. The TOE inspects each packet header and payload for anomalies and known signature-based attacks and determines whether to allow traffic to traverse the TOE.

1.3.2.10 Stateful Traffic Filtering and Packet Filtering

The TOE restricts the flow of network traffic between protected networks and other attached networks based on addresses and ports of the network nodes originating (source) and/or receiving (destination) applicable network traffic, as well as on established connection information.

The TOE performs packet filtering on network packets.

1.3.3 TOE Documentation

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

- **Sonicwall SonicOS/X v7.0.1 with VPN and IPS on TZ, NSa, NSsp, and NSv Appliances Security Target, version 1.2**
- **Sonicwall SonicOS 7.0.1 Common Criteria Administration Guide for NDPP, December 2024**

1.4 TOE Environment

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

- **Management workstation.** Any IT environment management workstation.
- **Remote Logging.** Audit Server supporting syslog protocol with an IPsec peer supporting IKEv2 and ESP.
- **Management Console.** Any computer that provides a supported browser to access administrative web GUI via HTTPS and direct serial connection providing administrative CLI access.
- **VPN Gateway.** VPN connections via IPSec.
- **WAN/Internet.** External IP interface.
- **LAN/Internal.** Internal IP interface.

1.5 Product Functionality not Included in the Scope of the Evaluation

The following product functionality is not included in the CC evaluation:

- Although SonicWall SonicOS Enhanced supports several authentication mechanisms, the following mechanisms are excluded from the evaluated configuration:
    - Remote Authentication Dial-In User Service (RADIUS)
    - Lightweight Directory Access Protocol (LDAP)
    - Active Directory (AD)
    - eDirectory authentication
- Command Line Interface (CLI) (Secure Shell (SSH))
- Hardware Failover
- Real-time Blacklist (Simple Mail Transfer Protocol (SMTP))
- Global Security Client (including Group VPN)
- Global Management System
- SonicPoint
- Voice over IP (VoIP)
- Network Time Protocol (NTP)
- Antivirus
- Application Firewall

2 Conformance Claims

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

2.1 CC Conformance Claims

The TOE is conformant to the following:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 5, April 2017 (Extended)
- Common Criteria for Information Technology Security Evaluations Part 3, Version 3.1, Revision 5, April 2017 **(**Conformant**)**

2.2 Protection Profile Conformance

This ST claims exact conformance to the following:

- **(NDcPP + IPS MOD + FW +VPNGW)** PP-Configuration for Network Device, Intrusion Prevention Systems, Stateful Traffic Filter Firewalls, and Virtual Private Network Gateways, Version 1.2

This PP-Configuration includes the following:

- collaborative Protection Profile for Network Devices, Version 2.2e (CPP_ND_V2.2E)
- PP-Module for Intrusion Protection Systems (IPS), Version 1.0 (MOD_IPS_V1.0)
- PP-Module for Stateful Traffic Filter Firewalls, Version 1.4 + Errata 20200625 (MOD_FW_1.4E)
- PP-Module for Virtual Private Network (VPN) Gateways, Version 1.3 (MOD_VPNGW_1.3)

2.3 Conformance Rationale

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

2.3.1 Technical Decisions

All NIAP TDs issued to date and applicable to NDcPP v2.2e have been considered. The following tables identify all applicable TDs.

Table 5 – Relevant Technical Decisions (CPP_ND)

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD0527: Updated to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	
TD0528: NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	No	The ST does not claim NTP functionality.
TD0536: NIT Technical Decision for Update Verification Inconsistency	Yes	
TD0537: NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	No	The ST does not claim TLSC functionality.
TD0546: NIT Technical Decision for DTLS – clarification of Application Note 63	No	The ST does not claim DTLSC functionality.
TD0547: NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes	
TD0555: NIT Technical Decision for RFC Reference incorrect in TLSS Test	Yes	
TD0556: NIT Technical Decisions for RFC 5077 question	Yes	
TD0563: NIT Technical Decision for Clarification of audit date information	Yes	
TD0564: NIT Technical Decision for Vulnerability Analysis Search Criteria	Yes	
TD0569: NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	Yes	
TD0570: NIT Technical Decision for Clarification about FIA_AFL.1	Yes	
TD0571: NIT Technical Decision for Guidance on how to handle FIA_AFL.1	Yes	
TD0572: NIT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes	
TD0580: NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	Yes	
TD0581: NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	Yes	
TD0591: NIT Technical Decision for Virtual TOEs and hypervisors	Yes	
TD0592: NIT Technical Decision for Local Storage of Audit Records	Yes	

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

Table 6 – Relevant Technical Decisions (MOD_CPP_FW)

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD0545: NIT Technical Decision for Conflicting FW rules cannot be configured (extension of RfI#201837)	Yes	
TD0551: NIT Technical Decision for Incomplete Mappings of OEs in FW Module v1.4+Errata	Yes	
TD0827: Aligning MOD_CPP_FW_v1.4E with CPP_ND_V3.0E	No	This TD modifies the text in the MOD_VPNGW_V1.4e and MOD_VPNGW_V1.4e-SD but the update is to include the CPP_ND_V3.0E which is not applicable for this evaluation.

Table 7 – Relevant Technical Decisions (MOD_VPNGW)

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD0781: Correction to FIA_PSK_EXT.3 EA for MOD_VPNGW_v1.3	No	FIA_PSK_EXT.3 is not claimed
TD0811: Correction to Referenced SFR in FIA_PSK_EXT.3 Test	No	FIA_PSK_EXT.3 is not claimed
TD0824: Aligning MOD_VPNGW 1.3 with NDcPP 3.0E	Yes	This TD modifies the text in the MOD_VPNGW_V1.3 and MOD_VPNGW_V1.3-SD to include the CPP_ND_V3.0E which is not applicable to this evaluation, but the TD also modifies the application notes and selections for some FCS_IPSEC_EXT.1 SFRs which is applicable to the evaluation.
TD0838: PPK Configurability in FIA_PSK_EXT.1.1	No	FIA_PSK_EXT.1 is not claimed

Table 8 – Relevant Technical Decisions (MOD_IPS)

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD0595: Administrative corrections to IPS PP-Module	Yes	
TD0722: IPS_SBD_EXT.1.1 EA Correction	Yes	
TD0828: Aligning MOD_IPS_V1.0 with CPP_ND_V3.0E	No	This TD modifies the text in the MOD_IPS_V1.0 and MOD_IPS_V1.1-SD but the update is to include the CPP_ND_V3.0E which is not applicable for this evaluation.

3 Security Problem Definition

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

3.1 Threats

The threats included in Table 9 to 12 are drawn directly from the PP and the Modules specified in Section 2.2.

Table 9 – Threats (CPP_ND)

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

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

Table 10 – Threats (MOD_CPP_FW)

ID	Threat
T.NETWORK_DISCLOSURE	An attacker may attempt to "map" a subnet to determine the machines that reside on the network, and obtaining the IP addresses of machines, as well as the services (ports) those machines are offering. This information could be used to mount attacks to those machines via the services that are exported.

ID	Threat
T.NETWORK_ACCESS	With knowledge of the services that are exported by machines on a subnet, an attacker may attempt to exploit those services by mounting attacks against those services.
T.NETWORK_MISUSE	An attacker may attempt to use services that are exported by machines in a way that is unintended by a site's security policies. For example, an attacker might be able to use a service to "anonymize" the attacker's machine as they mount attacks against others.
T.MALICIOUS_TRAFFIC	An attacker may attempt to send malformed packets to a machine in hopes of causing the network stack or services listening on UDP/TCP ports of the target machine to crash.

Table 11 – Threats (MOD_VPNGW)

ID	Threat
T.DATA_INTEGRITY	Devices on a protected network may be exposed to threats presented by devices located outside the protected network that may attempt to modify the data without authorization. If known malicious external devices are able to communicate with devices on the protected network or if devices on the protected network can communicate with those external devices then the data contained in the communications may be susceptible to a loss of integrity.
T.NETWORK_ACCESS	Devices located outside the protected network may seek to exercise services located on the protected network that are intended to only be accessed from inside the protected network or only accessed by entities using an authenticated path into the protected network. Devices located outside the protected network may, likewise, offer services that are inappropriate for access from within the protected network. From an ingress perspective, VPN gateways can be configured so that only those network servers intended for external consumption by entities operating on a trusted network (e.g., machines operating on a network where the peer VPN gateways are supporting the connection) are accessible and only via the intended ports. This serves to mitigate the potential for network entities outside a protected network to access network servers or services intended only for consumption or access inside a protected network. From an egress perspective, VPN gateways can be configured so that only specific external services (e.g., based on destination port) can be accessed from within a protected network, or moreover are accessed via an encrypted channel. For example, access to external mail

ID	Threat
	services can be blocked to enforce corporate policies against accessing uncontrolled email servers, or, that access to the mail server must be done over an encrypted link.
T.NETWORK_DISCLOSURE	Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to conduct unauthorized activities. If known malicious external devices are able to communicate with devices on the protected network, or if devices on the protected network can establish communications with those external devices (e.g., as a result of a phishing episode or by inadvertent responses to email messages), then those internal devices may be susceptible to the unauthorized disclosure of information. From an infiltration perspective, VPN gateways serve not only to limit access to only specific destination network addresses and ports within a protected network, but whether network traffic will be encrypted or transmitted in plaintext. With these limits, general network port scanning can be prevented from reaching protected networks or machines, and access to information on a protected network can be limited to that obtainable from specifically configured ports on identified network nodes (e.g., web pages from a designated corporate web server). Additionally, access can be limited to only specific source addresses and ports so that specific networks or network nodes can be blocked from accessing a protected network thereby further limiting the potential disclosure of information. From an exfiltration perspective, VPN gateways serve to limit how network nodes operating on a protected network can connect to and communicate with other networks limiting how and where they can disseminate information. Specific external networks can be blocked altogether or egress could be limited to specific addresses or ports. Alternately, egress options available to network nodes on a protected network can be carefully managed in order to, for example, ensure that outgoing connections are encrypted to further mitigate inappropriate disclosure of data through packet sniffing.
T.NETWORK_MISUSE	Devices located outside the protected network, while permitted to access particular public services offered inside the protected network, may attempt to conduct inappropriate activities while communicating with those allowed public services. Certain services offered from within a protected network may also represent a risk when accessed from outside the protected network. From an ingress perspective, it is generally assumed that entities operating on external networks are not bound by the use policies for a given protected network.

ID	Threat
	Nonetheless, VPN gateways can log policy violations that might indicate violation of publicized usage statements for publicly available services. From an egress perspective, VPN gateways can be configured to help enforce and monitor protected network use policies. As explained in the other threats, a VPN gateway can serve to limit dissemination of data, access to external servers, and even disruption of services – all of these could be related to the use policies of a protected network and as such are subject in some regards to enforcement. Additionally, VPN gateways can be configured to log network usages that cross between protected and external networks and as a result can serve to identify potential usage policy violations.
T.REPLAY_ATTACK	If an unauthorized individual successfully gains access to the system, the adversary may have the opportunity to conduct a "replay" attack. This method of attack allows the individual to capture packets traversing throughout the network and send the packets at a later time, possibly unknown by the intended receiver. Traffic is subject to replay if it meets the following conditions: • Cleartext: an attacker with the ability to view unencrypted traffic can identify an appropriate segment of the communications to replay as well in order to cause the desired outcome • No integrity: alongside cleartext traffic, an attacker can make arbitrary modifications to captured traffic and replay it to cause the desired outcome if the recipient has no means to detect these

Table 12 – Threats (MOD_IPS)

ID	Threat
T. NETWORK_ACCESS	Unauthorized access may be achieved to services on a protected network from outside that network, or alternately services outside a protected network from inside the protected network. If malicious external devices are able to communicate with devices on the protected network via a backdoor then those devices may be susceptible to the unauthorized disclosure of information.
T.NETWORK_DISCLOSURE	Sensitive information on a protected network might be disclosed resulting from ingress- or egress-based actions.
T.NETWORK_DOS	Attacks against services inside a protected network, or indirectly by virtue of access to malicious agents from within a protected network, might lead to denial of services otherwise available within a protected network.

ID	Threat
T.NETWORK_MISUSE	Access to services made available by a protected network might be used counter to operational environment policies. Devices located outside the protected network may attempt to conduct inappropriate activities while communicating with allowed public services. (e.g. manipulation of resident tools, SQL injection, phishing, forced resets, malicious zip files, disguised executables, privilege escalation tools and botnets).

3.2 Assumptions

The assumptions included in the Table 13-15 are drawn directly from PP and the relevant Modules.

Table 13 – Assumptions (CPP_ND)

ID	Assumption
A.PHYSICAL_PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). 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.

ID	Assumption
A.NO_THRU_TRAFFIC_PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall).
A.TRUSTED_ADMINISTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. (The paragraph that follows is for x509v3 cert-based authentication. If not relevant, remove) 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.COMPONENTS_RUNNING (applies to distributed TOEs only)	For distributed TOEs it is assumed that the availability of all TOE components is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. It is also assumed that in addition to the availability of all components it is also checked as appropriate that the audit functionality is running properly on all TOE components.

ID	Assumption
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 (applies to vNDs only)	The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.
A.VS_REGULAR_UPDATES (applies to vNDs only)	The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.VS_ISOLATION (applies to vNDs only)	For vNDs, it is assumed that the VS provides, and is configured to provide sufficient isolation between software running in VMs on the same physical platform. Furthermore, it is assumed that the VS adequately protects itself from software running inside VMs on the same physical platform.
A.VS_CORRECT_CONFIGURATION (applies to vNDs only)	For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.

Table 14 – Assumptions (MOD_VPNGW)

ID	Assumption
A.CONNECTIONS	This assumption defines the TOE's placement in a network such that it is able to perform its required security functionality. The Base-PP does not define any assumptions about the TOE's architectural deployment so there is no conflict here.

Table 15 – Assumptions (MOD_IPS)

ID	Assumption
A.CONNECTIONS	It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE's security policies will be enforced on all applicable network traffic flowing among the attached networks.

3.3 Organizational Security Policies

The OSPs included in Table 1 and 17 are drawn directly from the PP and any relevant Modules.

Table 16 – OSPs (CPP_ND)

ID	OSP
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

Table 17 – OSPs (MOD_IPS)

ID	OSP
P.ANALYZE	Analytical processes and information to derive conclusions about potential intrusions must be applied to IPS data and appropriate response actions taken.

4 Security Objectives

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

4.1 Security Objectives for the TOE

The security objectives in the following tables apply to the TOE.

Table 18 – Security Objectives for the TOE (MOD_CPP_FW)

ID	Security Objectives
O.RESIDUAL_ INFORMATION	The TOE shall implement measures to ensure that any previous information content of network packets sent through the TOE is made unavailable either upon deallocation of the memory area containing the network packet or upon allocation of a memory area for a newly arriving network packet or both.
O.STATEFUL_TRAFFIC_ FILTERING	The TOE shall perform stateful traffic filtering on network packets that it processes. For this the TOE shall support the definition of stateful traffic filtering rules that allow to permit or drop network packets. The TOE shall support assignment of the stateful traffic filtering rules to each distinct network interface. The TOE shall support the processing of the applicable stateful traffic filtering rules in an administratively defined order. The TOE shall deny the flow of network packets if no matching stateful traffic filtering rule is identified. Depending on the implementation, the TOE might support the stateful traffic filtering of Dynamic Protocols (optional).

Table 19 – Security Objectives for the TOE (MOD_VPNGW)

ID	Security Objectives
O.ADDRESS_FILTERING	To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption or denial of services, and network-based reconnaissance, compliant TOE's will implement packet filtering capability. That capability will restrict the flow of network traffic between protected networks and other attached networks based on network addresses of the network nodes originating (source) or receiving (destination) applicable network traffic as well as on established connection information.
O.AUTHENTICATION	To further address the issues associated with unauthorized disclosure of information, a compliant TOE's authentication ability (IPSec) will allow a VPN peer to establish VPN connectivity with another VPN

ID	Security Objectives
	peer and ensure that any such connection attempt is both authenticated and authorized. VPN endpoints authenticate each other to ensure they are communicating with an authorized external IT entity.
O.CRYPTOGRAPHIC_ FUNCTIONS	To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption of services, and network-based reconnaissance, compliant TOE's will implement cryptographic capabilities. These capabilities are intended to maintain confidentiality and allow for detection and modification of data that is transmitted outside of the TOE.
O.FAIL_SECURE	There may be instances where the TOE's hardware malfunctions or the integrity of the TOE's software is compromised, the latter being due to malicious or non-malicious intent. To address the concern of the TOE operating outside of its hardware or software specification, the TOE will shut down upon discovery of a problem reported via the self-test mechanism and provide signature-based validation of updates to the TSF.
O.PORT_FILTERING	To further address the issues associated with unauthorized disclosure of information, etc., a compliant TOE's port filtering capability will restrict the flow of network traffic between protected networks and other attached networks based on the originating (source) or receiving (destination) port (or service) identified in the network traffic as well as on established connection information.
O.SYSTEM_MONITORING	To address the issues of administrators being able to monitor the operations of the VPN gateway, it is necessary to provide a capability to monitor system activity. Compliant TOEs will implement the ability to log the flow of network traffic. Specifically, the TOE will provide the means for administrators to configure packet filtering rules to 'log' when network traffic is found to match the configured rule. As a result, matching a rule configured to 'log' will result in informative event logs whenever a match occurs. In addition, the establishment of security associations (SAs) is auditable, not only between peer VPN gateways, but also with certification authorities (CAs).
O.TOE_ADMINISTRATION	TOEs will provide the functions necessary for an administrator to configure the packet filtering rules, as well as the cryptographic aspects of the IPsec protocol that are enforced by the TOE.

Table 20 – Security Objectives for the TOE (MOD_IPS)

ID	Security Objectives
O.SYSTEM_MONITORING	To be able to analyze and react to potential network policy violations, the IPS must be able to collect and store essential data elements of network traffic on monitored networks.
O.IPS_ANALYZE	Entities that reside on or communicate across monitored networks must have network activity effectively analyzed for potential violations of approved network usage. The TOE must be able to effectively analyze data collected from monitored networks to reduce the risk of unauthorized disclosure of information, inappropriate access to services, and misuse of network resources.
O.IPS_REACT	The TOE must be able to react in real-time as configured by the Security Administrator to terminate and block traffic flows that have been determined to violate administrator-defined IPS policies.
O.TOE_ADMINISTRATION	To address the threat of unauthorized administrator access that is defined in the Base-PP, conformant TOEs will provide the functions necessary for an administrator to configure the IPS capabilities of the TOE.

4.2 Security Objectives for the Operational Environment

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

Table 21 – Security Objectives for the Operational Environment (CPP_ND)

ID	Objectives for the Operational Environment
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

ID	Objectives for the Operational Environment
OE.TRUSTED_ADMN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.COMPONENTS_RUNNING (applies to distributed TOEs only)	For distributed TOEs, the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.
OE.VM_CONFIGURATION (applies to vNDs only)	For vNDs, the Security Administrator ensures that the VS and VMs are configured to • Reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and • Correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting).

Table 22 – Security Objectives for the Operational Environment (MOD_CPP_FW)

ID	Objectives for the Operational Environment
All Security Objectives for the Operational Environment of the CPP_ND identified in table 21 are applicable	All objectives for the Operational Environment of the Base-PP apply also to this PP-Module. OE.NO_THRU_TRAFFIC_PROTECTION is still operative, but only for the interfaces in the TOE that are defined by the Base-PP and not the PP-Module.

Table 23 – Security Objectives for the Operational Environment (MOD_VPNGW)

ID	Objectives for the Operational Environment
OE.CONNECTIONS	The TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

Table 24 – Security Objectives for the Operational Environment (MOD_IPS)

ID	Objectives for the Operational Environment
OE.CONNECTIONS	TOE administrators will ensure that the TOE is installed in a manner that will allow the TOE to effectively enforce its policies on network traffic of monitored networks.

5 Security Requirements

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

Table 25 – SFRs

Requirement	Description
FAU_GEN.1	Audit Data Generation
FAU_GEN.1/VPN	Audit Data Generation (VPN)
FAU_GEN.1/IPS	Audit Data Generation (IPS)
FAU_GEN.2	User Identity Association
FAU_STG_EXT.1	Protected Audit Event Storage
FCS_CKM.1	Cryptographic Key Generation
FCS_CKM.1/IKE	Cryptographic Key Generation
FCS_CKM.2	Cryptographic Key Establishment
FCS_CKM.4	Cryptographic Key Destruction
FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
FCS_HTTPS_EXT.1	HTTPS Protocol
FCS_IPSEC_EXT.1	IPsec Protocol
FCS_RBG_EXT.1	Random Bit Generation
FCS_TLSS_EXT.1	TLS Server Protocol
FDP_RIP.2	Full residual information protection
FIA_AFL.1	Authentication Failure Management
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU_EXT.2	Password-based Authentication Mechanism
FIA_UAU.7	Protected Authentication Feedback
FIA_X509_EXT.1/Rev	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FIA_X509_EXT.3	X.509 Certificate Requests
FMT_MOF.1/ManualUpdate	Management of Security Functions Behaviour
FMT_MOF.1/Services	Management of Security Functions Behaviour
FMT_MTD.1/CoreData	Management of TSF Data
FMT_MTD.1/CryptoKeys	Management of TSF Data
FMT_SMF.1	Specification of Management Functions

Requirement	Description
FMT_SMF.1/FFW	Specification of Management Functions (Firewall)
FMT_SMF.1/VPN	Specification of Management Functions (VPN Gateway)
FMT_SMF.1/IPS	Specification of Management Functions (IPS)
FMT_SMR.2	Restrictions on security roles
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_FLS.1/SelfTest	SelfTest Fail Secure
FPT_STM_EXT.1	Reliable Time Stamps
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
FPT_TST_EXT.1	TSF Testing
FPT_TST_EXT.3	TSF Testing (Extended)
FPT_TUD_EXT.1	Trusted Update
FTA_SSL_EXT.1	TSF-initiated Session Locking
FTA_SSL.3	TSF-initiated Termination
FTA_SSL.4	User-initiated Termination
FTA_TAB.1	Default TOE Access Banner
FTP_ITC.1	Inter-TSF Trusted Channel
FTP_ITC.1/VPN	Inter-TSF Trusted Channel (VPN Communications)
FTP_TRP.1/Admin	Admin Trusted Path
FFW_RUL_EXT.1	Stateful Traffic Filtering
FPF_RUL_EXT.1	Rules for Packet Filtering
IPS_ABD_EXT.1	Anomaly-Based IPS Functionality
IPS_IPB_EXT.1	IP Blocking
IPS_NTA_EXT.1	Network Traffic Analysis
IPS_SBD_EXT.1	Signature-Based IPS Functionality

5.1 Conventions

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

- Unaltered SFRs are stated in the form used in [CC2] or their extended component definition (ECD);
- Refinement made in the PP: the refinement text is indicated with **bold text** and ~~strikethroughs~~;
- Selection wholly or partially completed in the PP: the selection values (i.e. the selection values adopted in the PP or the remaining selection values available for the ST) are indicated with underlined text
  e.g. '[selection: *disclosure, modification, loss of use*]' in [CC2] or an ECD might become 'disclosure' (completion) or '[selection: disclosure, modification]' (partial completion) in the PP;
- Assignment wholly or partially completed in the PP: indicated with *italicized text*;
- Assignment completed within a selection in the PP: the completed assignment text is indicated with *italicized and underlined text*

e.g. '[selection: *change_default, query, modify, delete, [assignment: other operations]*]' in [CC2] or an ECD might become 'change_default, select_tag' (completion of both selection and assignment) or '[selection: change_default, select_tag, select_value]' (partial completion of selection, and completion of assignment) in the PP;

- Iteration: indicated by adding a string starting with '/' (e.g. 'FCS_COP.1/Hash').
- Extended SFRs are identified by having a label 'EXT' at the end of the SFR name.

5.2 Security Functional Requirements

This section includes the security functional requirements for this ST.

5.2.1 Security Audit (FAU)

5.2.1.1 FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1

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

a) Start-up and shut-down of the audit functions;
b) 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 26.*

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 .

Table 26 – Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG_EXT.1	None	None
FCS_CKM.1	None	None
FCS_CKM.2	None	None
FCS_CKM.4	None	None

Requirement	Auditable Events	Additional Audit Record Contents
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_HTTPS_EXT.1	Failure to establish a HTTPS Session	Reason for failure
FCS_IPSEC_EXT.1	Failure to establish an IPSec SA.	Reason for failure.
FCS_RBG_EXT.1	None	None
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None	None
FIA_UIA_EXT.1	All use of identification and authentication mechanism	Origin of the attempt (e.g., IP address)
FIA_UAU_EXT.2	All use of identification and authentication mechanism	Origin of the attempt (e.g., IP address)
FIA_UAU.7	None	None
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None	None
FIA_X509_EXT.3	None	None
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None
FMT_MOF.1/Services	None	None
FMT_MTD.1/CoreData	None	None
FMT_MTD.1/CryptoKeys	None	None
FMT_SMF.1	All management activities of TSF data	None
FMT_SMF.1/FFW	All management activities of TSF data (including creation, modification and deletion of firewall rules).	None
FMT_SMR.2	None	None
FPT_SKP_EXT.1	None	None
FPT_APW_EXT.1	None	None
FPT_TST_EXT.1	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None
FTA_SSL.3	The termination of a remote session by the session locking mechanism	None
FTA_SSL.4	The termination of an interactive session	None
FTA_SSL_EXT.1 (if “terminate the session” is selected)	The termination of a local session by the session locking mechanism	None
FTA_TAB.1	None	None
FTP_ITC.1	• Initiation of the trusted channel • Termination of the trusted channel • Failure of the trusted channel functions	Identification of the initiator and target of failed trusted channels establishment attempt
FTP_TRP.1/Admin	• Initiation of the trusted path • Termination of the trusted path. • Failure of the trusted path functions.	None
FDP_RIP.2	None	None
FFW_RUL_EXT.1	Application of rules configured with the ‘log’ operation	• Source and destination addresses • Source and destination ports • Transport Layer Protocol • TOE Interface

5.2.1.2 FAU_GEN.1/VPN Audit Data Generation (VPN Gateway)

FAU_GEN.1.1/VPN

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) Indication that TSF self-test was completed
c) Failure of self-test
d) Auditable events for the [*not specified*] level of audit; and
e) *[auditable events defined in the Auditable Events for Mandatory Requirements table].*

FAU_GEN.1.2/VPN

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 PP/ST, *[additional information defined in the Auditable Events for Mandatory Requirements table for each auditable event, where applicable].*

Table 27 – Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1/VPN	No events specified.	N/A
FCS_CKM.1/IKE	No events specified.	N/A
FMT_SMF.1//VPN	All administrative actions.	No additional information.
FPF_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport layer protocol
FPT_FLS.1/SelfTest	No events specified.	N/A
FPT_TST_EXT.3	No events specified.	N/A
FTP_ITC.1/VPN	Initiation of the trusted channel	No additional information.
FTP_ITC.1/VPN	Termination of the trusted channel	No additional information.
FTP_ITC.1/VPN	Failure of the trusted channel functions	Identification of the initiator and target of failed trusted channel establishment attempt

5.2.1.3 FAU_GEN.1/IPS Audit Data Generation for IPS Refinement

FAU_GEN.1.1/IPS

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

a) Start-up and shut-down of the **IPS** functions;
b) All **IPS** auditable events for the [*not specified*] level of audit; and
c) [*All dissimilar IPS events;*
d) *All dissimilar IPS reactions*;
e) *Totals of similar events occurring within a specified time period*;
f) *Totals of similar reactions occurring within a specified time period*;
g) *The events in the IPS Events table*.
h) *[no other auditable events*].

Application Note: This SFR has been updated as per TD0595.

FAU_GEN.1.2/IPS

The TSF shall record within each **IPS auditable event** record at least the following information:

a) Date and time of the event, type of event **and/or reaction**, ~~subject identity, and the outcome (success or failure) of the event;~~ and;
b) For each **IPS** audit**able** event type, based on the auditable event definitions of the functional components included in the PP~~/ST~~, [*information specified in column three of the IPS Events table*].

Table 28– Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1/IPS	No events specified.	N/A
FMT_SMF.1/IPS	Modification of an IPS policy element.	Identifier or name of the modified IPS policy element (e.g. which signature, baseline, or known-good/known-bad list was modified).
IPS_ABD_EXT.1	Inspected traffic matches an anomaly-based IPS policy.	Source and destination IP addresses.
		The content of the header fields that were determined to match the policy.
		TOE interface that received the packet.
		Aspect of the anomaly-based IPS policy rule that triggered the event (e.g. throughput, time of day, frequency, etc.).
		Network-based action by the TOE (e.g. allowed, blocked, sent reset to source IP, sent blocking notification to firewall).
IPS_IPB_EXT.1	Inspected traffic matches a list of known-good or known-bad addresses applied to an IPS policy.	Source and destination IP addresses (and, if applicable, indication of whether the source and/or destination address matched the list).
		TOE interface that received the packet.
		Network-based action by the TOE (e.g. allowed, blocked, sent reset).
IPS_NTA_EXT.1		Identification of the TOE interface.

Requirement	Auditable Events	Additional Audit Record Contents
	Modification of which IPS policies are active on a TOE interface. Enabling/disabling a TOE interface with IPS policies applied. Modification of which mode(s) is/are active on a TOE interface.	The IPS policy and interface mode (if applicable).
IPS_SBD_EXT.1	Inspected traffic matches a signature-based IPS rule with logging enabled.	Name or identifier of the matched signature.
		Source and destination IP addresses.
		The content of the header fields that were determined to match the signature.
		TOE interface that received the packet.
		Network-based action by the TOE (e.g. allowed, blocked, sent reset).

5.2.1.4 FAU_GEN.2 User Identity Association

FAU_GEN.2.1

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

5.2.1.5 FAU_STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1.1

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

FAU_STG_EXT.1.2

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

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

].

FAU_STG_EXT.1.3

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

5.2.2 Cryptographic Support (FCS)

5.2.2.1 FCS_CKM.1 Cryptographic Key Generation

FCS_CKM.1.1

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

- *RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;*
- *ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;*
- *FFC Schemes using 'safe-prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526*].

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

5.2.2.2 FCS_CKM.1/IKE Cryptographic Key Generation (for IKE Peer Authentication)

FCS_CKM.1.1

The TSF shall generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with a specified cryptographic key generation algorithm: [

FIPS PUB 186-4, "Digital Signature Standard (DSS)," Appendix B.3 for RSA schemes

FIPS PUB 186-4, "Digital Signature Standard (DSS)," Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-384 and [P-256, P-521]

] and [

FFC Schemes using "safe-prime" groups that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526]

] and specified cryptographic key sizes [equivalent to, or greater than, a symmetric key strength of 112 bits].

5.2.2.3 FCS_CKM.2 Cryptographic Key Establishment

FCS_CKM.2.1

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

- *RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1";*
- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";*
- *FFC Schemes using "safe-prime" groups that meet the following: 'NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment collaborative Protection*

Profile for Network Devices v2.2e, 23-March-2020 Page 57 of 174 Schemes Using Discrete Logarithm Cryptography" and [groups listed in RFC 3526].

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

Application Note: This SFR has been updated as per TD0580 and TD0581

5.2.2.4 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 [a pseudo-random pattern using the TSF's RBG]];*
- *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* [*a pseudorandom pattern using the TSF's RBG*];

    ]

that meets the following: *No Standard.*

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

FCS_COP.1.1/DataEncryption

The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [***CBC, GCM***] ***and*** **[*no other*]** mode and cryptographic key sizes [***128 bits, 256 bits***] **and [*192 bits*]** that meet the following: *AES as specified in ISO 18033-3,* [***CBC as specified in ISO 10116, GCM as specified in ISO 19772***] **and [*no other standards*]**.

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

FCS_COP.1.1/SigGen

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

- *RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048, 3072, 4096 bits]*
- *Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits, 384 bits, 521 bits]*

]

that meet the following: [

- *For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,*
- *For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384]; ISO/IEC 14888-3, Section 6.4*

].

5.2.2.7 FCS_COP.1/Hash Cryptographic Operations (Hash Algorithm)

FCS_COP.1.1/Hash

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

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

FCS_COP.1.1/KeyedHash

The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512*] and cryptographic key sizes *[160, 256, 384, 512]* **and message digest sizes [*160, 256, 384, 512*] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

5.2.2.9 FCS_HTTPS_EXT.1 HTTPS Protocol

FCS_HTTPS_EXT.1.1

The TSF shall implement the HTTPS protocol that complies with RFC 2818.

FCS_HTTPS_EXT.1.2

The TSF shall implement the HTTPS protocol using TLS.

FCS_HTTPS_EXT.1.3

If a peer certificate is presented, the TSF shall [*[not require client authentication]*] if the peer certificate is deemed invalid.

5.2.2.10 FCS_IPSEC_EXT.1 IPSec Protocol

FCS_IPSEC_EXT.1.1

The TSF shall implement the IPsec architecture as specified in RFC 4301.

FCS_IPSEC_EXT.1.2

The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched and discards it.

FCS_IPSEC_EXT.1.3

The TSF shall implement [*tunnel mode*].

FCS_IPSEC_EXT.1.4

The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms **[*AES-CBC-128 (RFC 3602), AES-CBC-256 (RFC 3602)*] *and* [*AES-CBC-192 (RFC 3602)*]** with a Secure Hash Algorithm (SHA)-based HMAC [*HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512]*.

FCS_IPSEC_EXT.1.5

The TSF shall implement the protocol: [

- *IKEv2 as defined in RFC 5996 and [with mandatory support for NAT traversal as specified in RFC 5996, section 2.23)], and [RFC 4868 for hash functions]*

].

FCS_IPSEC_EXT.1.6

The TSF shall ensure the encrypted payload in the [*IKEv2*] protocol uses the cryptographic algorithms [*AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 5282)*].

FCS_IPSEC_EXT.1.7

The TSF shall ensure that [

- *IKEv2 SA lifetimes can be configured by a Security Administrator based on*

  [

  - *length of time, where the time values can be configured within [2 minutes to 24] hours*

  ]

].

FCS_IPSEC_EXT.1.8

The TSF shall ensure that [

- *IKEv2 Child SA lifetimes can be configured by a Security Administrator based on*

  [

  - *length of time, where the time values can be configured within [2 minutes to 8] hours;*

  ]

].

FCS_IPSEC_EXT.1.9

The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [224, 256, *384, 512]* bits.

FCS_IPSEC_EXT.1.10

The TSF shall generate nonces used in [*IKEv2*] exchanges of length [

- *at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash*

  ].

FCS_IPSEC_EXT.1.11

The TSF shall ensure that IKE protocols implement DH Group(s)

- **19 (256-bit Random ECP), 20 (384-bit Random ECP) according to RFC 5114 and**
- [*14 (2048-bit MODP)],*
- [*21 (521-bit Random ECP)] according to RFC 5114.*

].

FCS_IPSEC_EXT.1.12

The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [*IKEv2 IKE_SA*] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [*IKEv2 CHILD_SA*] connection.

FCS_IPSEC_EXT.1.13

The TSF shall ensure that all IKE protocols perform peer authentication using [*RSA, ECDSA*] that use X.509v3 certificates that conform to RFC 4945 and [*no other method*].

Application Note: This SFR has been updated as per TD0824.

FCS_IPSEC_EXT.1.14

The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: **Distinguished Name (DN*)***, [*SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN*].

5.2.2.11 FCS_RBG_EXT.1 Random Bit Generation

FCS_RBG_EXT.1.1

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

FCS_RBG_EXT.1.2

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

5.2.2.12 FCS_TLSS_EXT.1 TLS Sever 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_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*

- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*

] and no other ciphersuites.

FCS_TLSS_EXT.1.2

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

FCS_TLSS_EXT.1.3

The TSF shall perform key establishment for TLS using [*RSA with key size* [*2048 bits, 3072 bits, 4096 bits*], ECDHE curves [secp256r1, secp384r1, secp521r1] and no other curves].

FCS_TLSS_EXT.1.4

The TSF shall support [*session resumption based on session IDs according to RFC 4346 (TLS1.1) or RFC 5246 (TLS1.2), session resumption based on session tickets according to RFC 5077*].

5.2.3 Residual information protection (FDP)

5.2.3.1 FDP_RIP.2 Full Residual Information Protection

FDP_RIP.2.1

The TSF shall ensure that any previous information content of a resource is made unavailable upon the [deallocation of the resource from] all objects.

5.2.4 Identification and Authentication (FIA)

5.2.4.1 FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within *[1-99]* unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

FIA_AFL.1.2

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

5.2.4.2 FIA_PMG_EXT.1 Password Management

FIA_PMG_EXT.1.1

The TSF shall provide the following password management capabilities for administrative passwords:

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [*"!", "@", "#", "$", "%", "^", "&", "*", "(", ")"*]

b) Minimum password length shall be configurable to between *[15]* and *[99]* characters.

5.2.4.3 FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1.1

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

- Display the warning banner in accordance with FTA_TAB.1;
- [*access links to the Sonicwall knowledge-base websites*].

FIA_UIA_EXT.1.2

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

5.2.4.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2.1

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

5.2.4.5 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**.

5.2.4.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

FIA_X509_EXT.1.1/Rev

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

- RFC 5280 certificate validation and certification path validation **supporting a minimum path length of three certificates.**
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [*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 extendedKeyUsagefield.*

- *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

FIA_X509_EXT.1.2/Rev

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

5.2.4.7 FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

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

FIA_X509_EXT.2.2

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

Application Note: This SFR has been updated as per TD0537.

5.2.4.8 FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1

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

FIA_X509_EXT.3.2

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

5.2.5 Security Management (FMT)

5.2.5.1 FMT_MOF.1/ManualUpdate Management of Security Functions Behavior

FMT_MOF.1.1/ManualUpdate

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

5.2.5.2 FMT_MOF.1/Services Management of Security Functions Behaviour

FMT_MOF.1.1/Services

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

5.2.5.3 FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.1.1/CoreData

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

5.2.5.4 FMT_MTD.1/CryptoKeys Management of TSF Data

FMT_MTD.1.1/CryptoKeys

The TSF shall restrict the ability to *manage* the [*cryptographic keys* ***and certificates used for VPN operation***] to [*Security Administrators*].

5.2.5.5 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 and*** *[no other] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- ***Ability to manage the cryptographic keys;***
- ***Ability to configure the cryptographic functionality;***
- ***Ability to configure the lifetime for IPsec SAs;***

[

- *Ability to start and stop services;*
- *Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1;*
- *Ability to set the time which is used for time-stamps;*
- *Ability to configure the reference identifier for the peer;*
- *Ability to import X.509v3 certificates to the TOE's trust store;*

].

5.2.5.6 FMT_SMF.1/FFW Specification of Management Functions (Firewall)

FMT_SMF.1.1/FFW

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

- *Ability to configure firewall rules;*

5.2.5.7 FMT_SMF.1/VPN Specification of Management Functions (VPN Gateway)

FMT_SMF.1.1/VPN

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

- *Definition of packet filtering rules*
- *Association of packet filtering rules to network interfaces*
- *Ordering of packet filtering rules by priority*

[

- *No other capabilities*

]].

5.2.5.8 FMT_SMF.1/IPS Specification of Management Functions (IPS)

FMT_SMF.1.1/IPS

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

- *Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality*
- *Modify these parameters that define the network traffic to be collected and analyzed:*
    - *Source IP addresses (host address and network address)*
    - *Destination IP addresses (host address and network address)*
    - *Source port (TCP and UDP)*
    - *Destination port (TCP and UDP)*
    - *Protocol (IPv4 and IPv6)*
    - *ICMP type and code*
- *Update (import) signatures*
- *Create custom signatures*
- *Configure anomaly detection*
- *Enable and disable actions to be taken when signature or anomaly matches are detected*
- *Modify thresholds that trigger IPS reactions*
- *Modify the duration of traffic blocking actions*
- *Modify the known-good and known-bad lists (of IP addresses or address ranges)*
- *Configure the known-good and known-bad lists to override signature-based IPS policies].*

5.2.5.9 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1

The TSF shall maintain the roles:

- *Security Administrator*

FMT_SMR.2.2

The TSF shall be able to associate users with roles.

FMT_SMR.2.3

The TSF shall ensure that the conditions

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

are satisfied.

5.2.6 Protection of the TSF (FPT)

5.2.6.1 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.

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

FPT_SKP_EXT.1.1

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

5.2.6.3 FPT_STM_EXT.1 Reliable Time Stamps

FPT_STM_EXT.1.1

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

FPT_STM_EXT.1.2

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

5.2.6.4 FPT_TST_EXT.1 TSF Testing

FPT_TST_EXT.1.1

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

- *Appliance Power on self-test consisting of a CPU and RAM test*
- *Firmware integrity test*
- *AES-CBC/AES-GCM Encrypt and Decrypt Known Answer Tests*
- *SHA-1, -256, -384, -512 Known Answer Tests*
- *HMAC-SHA-1, -256, -512 Known Answer Tests*
- *DSA Signature Verification Pairwise Consistency Test*
- *RSA Sign and Verify Known Answer Tests*
- *DH Pairwise Consistency Test*
- *DRBG Known Answer Test*
- *ECDSA Known Answer Test*
- *ECSDA Signature and Verification Known Answer Tests*

].

5.2.6.5 FPT_TST_EXT.3 TSF Self-Test with Defined Methods

FPT_TST_EXT.3.1

The TSF shall run a suite of the following self-tests [*[when loaded for execution]*] to demonstrate the correct operation of the TSF: [*integrity verification of stored executable code*].

FPT_TST_EXT.3.2

The TSF shall execute the self-testing through [a TSF-provided cryptographic service specified in FCS_COP.1/**SigGen**].

5.2.6.6 FPT_TUD_EXT.1 Trusted Update

FPT_TUD_EXT.1.1

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

FPT_TUD_EXT.1.2

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

FPT_TUD_EXT.1.3

The TSF shall provide means to authenticate firmware/software updates to the TOE using a **digital signature mechanism and** [***no other mechanisms***] prior to installing those updates.

5.2.6.7 FPT_FLS.1/SelfTest Fail Secure (Self-Test Failures)

FPT_FLS.1.1/SelfTest

The TSF shall **shut down** when the following types of failures occur: [failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests].

5.2.7 TOE Access (FTA)

5.2.7.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1

The TSF Shall, for local interactive sessions, [

- *terminate the session*]

after a Security Administrator-specified time period of inactivity

5.2.7.2 FTA_SSL.3 TSF-initiated Termination

FTA_SSL.3.1

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

5.2.7.3 FTA_SSL.4 User-initiated Termination

FTA_SSL.4.1

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

5.2.7.4 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.

5.2.8 Trusted Path/Channels (FTP)

5.2.8.1 FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1

The TSF shall **be capable of using [*IPsec*] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [*[VPN communications]*]** 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 *[transmission of audit data]*.

5.2.8.2 FTP_ITC.1/VPN Inter-TSF Trusted Channel (VPN Communications)

FTP_ITC.1.1/VPN

The TSF shall **be capable of using IPsec to** provide a trusted communication channel between itself and **authorized IT entities supporting VPN communications** 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/VPN

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

FTP_ITC.1.3/VPN

The TSF shall initiate communication via the trusted channel for [*remote VPN gateways or peers*].

5.2.8.3 FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin

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

FTP_TRP.1.2/Admin

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

FTP_TRP.1.3/Admin

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

5.2.9 Stateful Traffic Filter Firewall (FFW)

5.2.9.1 FFW_RUL_EXT.1 Stateful Traffic Filtering

FFW_RUL_EXT.1.1

The TSF shall perform stateful traffic filtering on network packets processed by the TOE.

FFW_RUL_EXT.1.2

The TSF shall allow the definition of stateful traffic filtering rules using the following network protocol fields:

- ICMPv4
  - Type
  - Code
- ICMPv6
  - Type
  - Code
- IPv4
  - Source address
  - Destination Address
  - Transport Layer Protocol
- IPv6
  - Source address
  - Destination Address
  - Transport Layer Protocol
  - [*no other field*]
- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port

and distinct interface.

FFW_RUL_EXT.1.3

The TSF shall allow the following operations to be associated with stateful traffic filtering rules: permit or drop with the capability to log the operation.

FFW_RUL_EXT.1.4

The TSF shall allow the stateful traffic filtering rules to be assigned to each distinct network interface.

FFW_RUL_EXT.1.5

The TSF shall:

a) accept a network packet without further processing of stateful traffic filtering rules if it matches an allowed established session for the following protocols: TCP, UDP, [*no other protocols*] based on the following network packet attributes:
   1. TCP: source and destination addresses, source and destination ports, sequence number, Flags;

2. UDP: source and destination addresses, source and destination ports;
3. [*no other protocols*].

b) Remove existing traffic flows from the set of established traffic flows based on the following: [*session inactivity timeout, completion of the expected information flow*].

FFW_RUL_EXT.1.6

The TSF shall enforce the following default stateful traffic filtering rules on all network traffic:

a) The TSF shall drop and be capable of [*logging*] packets which are invalid fragments;
b) The TSF shall drop and be capable of [*logging*] fragmented packets which cannot be re-assembled completely;
c) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a broadcast network;
d) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a multicast network;
e) The TSF shall drop and be capable of logging network packets where the source address of the network packet is defined as being a loopback address;
f) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address "reserved for future use" (i.e. 240.0.0.0/4) as specified in RFC 5735 for IPv4;
g) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6;
h) The TSF shall drop and be capable of logging network packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified; and
i) [*no other rules*].

FFW_RUL_EXT.1.7

The TSF shall be capable of dropping and logging according to the following rules:

a) The TSF shall drop and be capable of logging network packets where the source address of the network packet is equal to the address of the network interface where the network packet was received;
b) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is a link-local address;
c) The TSF shall drop and be capable of logging network packets where the source address of the network packet does not belong to the networks associated with the network interface where the network packet was received.

FFW_RUL_EXT.1.8

The TSF shall process the applicable stateful traffic filtering rules in an administratively defined order.

FFW_RUL_EXT.1.9

The TSF shall deny packet flow if a matching rule is not identified.

FFW_RUL_EXT.1.10

The TSF shall be capable of limiting an administratively defined number of half-open TCP connections. In the event that the configured limit is reached, new connection attempts shall be dropped and the drop event shall be [*logged*].

5.2.10 Packet Filtering (FPF)

5.2.10.1 FPF_RUL_EXT.1 Rules for Packet Filtering

FPF_RUL_EXT.1.1

The TSF shall perform Packet Filtering on network packets processed by the TOE.

FPF_RUL_EXT.1.2

The TSF shall allow the definition of Packet Filtering rules using the following network protocols and protocol fields: [

- IPv4 (RFC 791)
  - source address
  - destination address
  - protocol
- IPv6 (RFC 8200)
  - source address
  - destination address
  - next header (protocol)
- TCP (RFC 793)
  - source port
  - destination port
- UDP (RFC 768)
  - source port
  - destination port

].

FPF_RUL_EXT.1.3

The TSF shall allow the following operations to be associated with packet filtering rules: permit and drop with the capability to log the operation.

FPF_RUL_EXT.1.4

The TSF shall allow the packet filtering rules to be assigned to each distinct network interface.

FPF_RUL_EXT.1.5

The TSF shall process the applicable packet filtering rules (as determined in accordance with FPF_RUL_EXT.1.4) in the following order: [*Administrator-defined*].

FPF_RUL_EXT.1.6

The TSF shall drop traffic if a matching rule is not identified.

5.2.11 Intrusion Prevention (IPS)

5.2.11.1 IPS_ABD_EXT.1 Anomaly-Based IPS Functionality

IPS_ABD_EXT.1.1

T The TSF shall support the definition of [*baselines ('expected and approved'), anomaly ('unexpected') traffic patterns*] including the specification of [

- *time of day*]

and the following network protocol fields:

- [*FTP commands, HTTP commands and content*]

IPS_ABD_EXT.1.2

The TSF shall support the definition of anomaly activity through [manual configuration by administrators].

IPS_ABD_EXT.1.3

The TSF shall allow the following operations to be associated with anomaly-based IPS policies:

- In any mode, for any sensor interface: [
  - *allow the traffic flow*]
- In inline mode:
  - [*allow the traffic flow*
  - *block/drop the traffic flow*
  - *and [no other actions]*]

5.2.11.2 IPS_IPB_EXT.1 IP Blocking

IPS_IPB_EXT.1.1

The TSF shall support configuration and implementation of known-good and known-bad lists of [*source, destination*] IP addresses and [*no additional address types*].

IPS_IPB_EXT.1.2

The TSF shall allow [*Security Administrators*] to configure the following IPS policy elements: [*IP addresses, no other IPS policy elements]*.

5.2.11.3 IPS_NTA_EXT.1 Network Traffic Analysis

IPS_NTA_EXT.1.1

The TSF shall perform analysis of IP-based network traffic forwarded to the TOE's sensor interfaces, and detect violations of administratively-defined IPS policies.

IPS_NTA_EXT.1.2

The TSF shall process (be capable of inspecting) the following network traffic protocols:

- [*Internet Protocol (IPv4), RFC 791*
- *Internet Protocol version 6 (IPv6), RFC 2460*
- *Internet control message protocol version 4 (ICMPv4), RFC 792*
- *Internet control message protocol version 6 (ICMPv6), RFC 2463*
- *Transmission Control Protocol (TCP), RFC 793*
- *User Data Protocol (UDP), RFC 768*].

IPS_NTA_EXT.1.3

The TSF shall allow the signatures to be assigned to sensor interfaces configured for promiscuous mode, and to interfaces configured for inline mode, and support designation of one or more interfaces as 'management' for communication between the TOE and external entities without simultaneously being sensor interfaces.

- Promiscuous (listen-only) mode: [*Gigabit Ethernet*];
- Inline (data pass-through) mode: [Gigabit *Ethernet*];
- Management mode: [Gigabit *Ethernet*];
- [
  - *no other interface types*].

5.2.11.4 IPS_SBD_EXT.1 Signature-Based IPS Functionality

IPS_SBD_EXT.1.1

The TSF shall support inspection of packet header contents and be able to inspect at least the following header fields: [

- IPv4: version; header Length; packet Length; ID; IP flags; fragment offset; time to live (TTL); protocol; header checksum; source address; destination address; IP options; and [*no other field*].
- IPv6: Version; payload length; next header; hop limit; source address; destination address; routing header; and [*traffic class, flow label*].
- ICMP: type; code; header checksum; and [*[Rest of Header (varies based on the ICMP type and code)]*].
- ICMPv6: type; code; and header checksum.
- TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.
- UDP: Source port; destination port; length; and UDP checksum].

IPS_SBD_EXT.1.2

The TSF shall support inspection of packet payload data and be able to inspect at least the following data elements to perform string-based pattern-matching: [

- ICMPv4 data: characters beyond the first 4 bytes of the ICMP header.
- ICMPv6 data: characters beyond the first 4 bytes of the ICMP header.
- TCP data (characters beyond the 20 byte TCP header), with support for detection of:
  i) *FTP (file transfer) commands: help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.*
  ii) *HTTP (web) commands and content: commands including GET and POST, and administrator defined strings to match URLs/URIs, and web page content.*
  iii) *SMTP (email) states: start state, SMTP commands state, mail header state, mail body state, abort state.*
  iv) *[no other types of TCP payload inspection];*
  - *UDP data: characters beyond the first 8 bytes of the UDP header;*
  - *[no other types of packet payload inspection]*].

IPS_SBD_EXT.1.3

The TSF shall be able to detect the following header-based signatures (using fields identified in IPS_SBD_EXT.1.1) at IPS sensor interfaces: [

a) *IP Attacks*
   i) *IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)*
   ii) *IP source address equal to the IP destination (Land attack)*
b) *ICMP Attacks*
   i) *Fragmented ICMP Traffic (e.g. Nuke attack)*
   ii) *Large ICMP Traffic (Ping of Death attack)*
c) *TCP Attacks*
   i) *TCP NULL flags*
   ii) *TCP SYN+FIN flags*
   iii) *TCP FIN only flags*
   iv) *TCP SYN+RST flags*
d) *UDP Attacks*
   i) *UDP Bomb Attack*
   ii) *UDP Chargen DoS Attack*].

IPS_SBD_EXT.1.4

The TSF shall be able to detect all the following traffic-pattern detection signatures, and to have these signatures applied to IPS sensor interfaces: [

a) *Flooding a host (DoS attack)*
   i) *ICMP flooding (Smurf attack, and ping flood)*
   ii) *TCP flooding (e.g. SYN flood)*
b) *Flooding a network (DoS attack)*
c) *Protocol and port scanning*
   i) *IP protocol scanning*
   ii) *TCP port scanning*
   iii) *UDP port scanning*
   iv) *ICMP scanning*].

IPS_SBD_EXT.1.5

The TSF shall allow the following operations to be associated with signature-based IPS policies:

- In any mode, for any sensor interface: [
    - *allow the traffic flow*]
- In inline mode:
    - block/drop the traffic flow;

    and [
    - *allow all traffic flow*].

IPS_SBD_EXT.1.6

The TSF shall support stream reassembly or equivalent to detect malicious payload even if it is split across multiple non-fragmented packets.

5.3 TOE SFR Dependencies Rationale for SFRs

The PP and any relevant Modules contain all the requirements claimed in this ST. As such, the dependencies are not applicable since the PP and the PP modules have been approved.

5.4 Security Assurance Requirements

The TOE assurance requirements for this ST are taken directly from the NDcPP and MOD_FW, MOD_VPNGW, MOD_IPS, which are derived from Common Criteria Version 3.1, Revision 5. The assurance requirements are summarized in Table .

Table 29 – Security Assurance Requirements

Assurance Class	Assurance Components	Component Description
Security Target	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_REQ.1	Stated security requirements
	ASE_SPD.1	Security problem definition
	ASE_TSS.1	TOE Summary Specification
Development	ADV_FSP.1	Basic functionality specification
Guidance Documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative Procedures
Life Cycle Support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests	ATE_IND.1	Independent testing – conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability survey

5.5 Assurance Measures

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

Table 30 – TOE Security Assurance Measures

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

6 TOE Summary Specification

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

Table 31 – TOE Summary Specification SFR Description

Requirement	TSS Description
FAU_GEN.1 FAU_GEN.1/IPS FAU_GEN.1/VPN FAU_GEN.2	The TOE generates audit records and stores them as management logs and user activity logs. The management logs record administrative logins and management activity, including changes to configuration and access control policies. User activity logs record blocked traffic, blocked websites, VPN activity and other events related to the firewall. Each record contains the date and time, event type, subject identity (when applicable) and outcome of the event. For events caused by a user, the identity of the user is included in the audit record. Each IPS event is recorded in the logs as a single event. (i.e. Multiple logs with similar events are never combined to create a more general log entry.) Each log entry is grouped in a log category based on event type. Logging can be enabled or disabled per category and event type. Authorized administrators can enable enhanced logging to record configuration changes to IPS functions. IPS audit records are generated with an ID, category, and priority that are specific to each event type. For example, a single IPS audit record for a TCP flood attack may include the following: • ID = 1366 • Category = Attack • Priority = ALERT • Message = TCP-Flooding machine %s blacklisted Contents of the audit records are described in the guidance document. This includes administrator login and management activities associated with cryptographic keys. The logs do not contain the cryptographic keys. The SonicWall device can be configured to log network traffic associated with the rules set for allowing or denying particular packets. To do this, the administrator performs the following steps: • Under System > Administration, go to 'Enhanced Audit Logging Support' and enable the associated checkbox • Go to Log > Settings • Go to Network > Network Access and find 'Packet Allowed'. Select the checkbox next to 'Display Events in Log Monitor' • Select 'Apply' All packets that enter the SonicWall device are assessed according to the configured rules. A log is created any time a packet is dropped because it does not match an access rule. If the interface is overwhelmed, the packet will be dropped even if it matches an access rule. The normal log entries associated with access rules are not made when packets are dropped due to an overwhelmed interface; instead log records that indicate packets

<table>
<tr><th>Requirement</th><th>TSS Description</th></tr>
<tr><td></td><td>where dropped on a specified interface because the interface was overwhelmed are generated.

The startup and shutdown of the audit function is tied to the startup and shutdown of the TOE and the TOE generates audit messages for this activity. In addition, when the self-tests are performed, audit logs for successful execution of individual tests are generated in addition to the audit log to indicate that all self-tests have passed. There are overall logs for successful and failure of the self-tests as well. When a self-test fails, the TOE enters into an error state and the local console provides an error message reflecting information about the specific failure to the security administrators.

In the case of key related operations, the name of the certificate the key is associated with is logged and used as the unique reference identifier.</td></tr>
<tr><td>FAU_STG_EXT.1</td><td>This SFR applies to the audit records for both FAU_GEN.1 and FAU_GEN.1/IPS.

When contained on the TOE, the logs are stored in a database file saved in a specifically reserved area of the System flash. Access to view these records is restricted to authorized administrators with the appropriate privilege from the WebGUI. Users who do not have the required privilege are not able to access the audit records. Administrators are not permitted to delete or modify the audit logs. The maximum number of audit log entries recorded in the database file is limited and this limit for each model is different.
<table>
<tr><th>Model</th><th>Maximum Log Entries</th></tr>
<tr><td>TZ270, TZ270W, TZ370, TZ370W, TZ470, TZ470W, TZ570, TZ570W, TZ570P</td><td>1000</td></tr>
<tr><td>TZ670</td><td>2000</td></tr>
<tr><td>NSa2700, NSa3700</td><td>5000</td></tr>
<tr><td>NSa4700</td><td>7000</td></tr>
<tr><td>NSa5700, NSa6700</td><td>10000</td></tr>
<tr><td>NSsp10700, NSsp11700, NSsp13700</td><td>10000</td></tr>
<tr><td>NSv270, NSv470, NSv870</td><td>10000</td></tr>
</table>
When the log entries capacity reaches 100%, the oldest 25% of log entries are automatically deleted to free up space for new entries. This setting is non-configurable.

In the evaluated configuration, the TOE is configured to send audit records to an audit server over an IPsec protected link. The link is established between the TOE and the audit server, and the records are sent over this</td></tr>
</table>

Requirement	TSS Description
	connection. For the exported audit logs, a separate buffer is maintained. The logs are sent continuously and are removed from the buffer as they are sent. If the connection to the audit server is lost, the logs are stored in a 32-kilobyte rolling log buffer. When the buffer becomes full, the oldest logs are overwritten. The audit logs are sent to the external audit server even when the local audit log database file is full.
FCS_CKM.1 FCS_CKM.1/IKE FCS_CKM.2	The TOE supports Rivest-Shamir-Adleman (RSA) using 2048-bits, 3072-bits, and 4096-bits keys, ECDSA using P-256 or P-384 or P-521 keys, and Diffie-Hellman Group 14. EC-Curves and RSA is used in support of TLS and IPsec. RSA and ECDSA keys are generated in accordance with FIPS PUB 186-4 as described in FCS_COP.1/SigGen. The TOE complies with the requirements in FIPS PUB 186-4, Appendix B.3 for RSA and FIPS PUB 186-4, Appendix B.4 for ECDSA. Diffie-Hellman Group 14 keys are generated using the parameters specified in RFC 3526 Section 3. The TOE performs Elliptic-Curve Diffie-Helman and Diffie-Hellman Group 14 to establish IPsec keys (FCS_IPSEC_EXT.1) for secure communications with VPN clients, VPN gateways, and the audit server. The TOE implements PKCS1_v1.5 conformant RSA-based key establishment scheme and NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" conformant EC-based key establishment scheme for asymmetric key establishment used in TLS (FCS_TLSS_EXT.1) for remote administration. The relevant CAVP certificate numbers are listed in Section 6.1.
FCS_CKM.4	Plaintext key materials held in volatile and non-volatile memory are zeroized after use by direct overwrite consisting of a pseudo-random pattern. The overwrites are read and verified. Section 6.2 below shows the origin, storage location and destruction details for all plaintext keys. Unless otherwise stated, the keys are generated by the TOE. The SonicWall key used to verify firmware updates supports ECDSA (P-256 NIST curve). The TOE includes two types of memory: RAM and flash. Ephemeral keys are only held in RAM, either in the System RAM or the RAM buffer. The RAM buffer is an area of the System RAM that is allocated for data storage for a period of time. Private keys are only held in plaintext in the RAM buffer. Private keys and public key certificates are stored encrypted in flash memory using OpenSSL. Private and public keys are overwritten in the RAM buffer after use. In the configuration file, only the sensitive data (password) is protected by using AES 256 hash. The encrypt key of this is hardcoded and saved in the flash memory. The initialization vector generated randomly to ensure randomness of the first block to ensure the protection of the key. Whenever the configuration file is updated, the initialization vector is also updated. When the configuration file is imported from outside, the TOE

Requirement	TSS Description
	generates a new initialization vector. When the TOE is rebooted, the initialization vector is refreshed. Setting the TOE to factory default zeroizes all keys, including the configuration file encrypting key and the keys stored in the flash memory.
FCS_COP.1/DataEncryption	The TOE provides AES encryption/decryption in CBC mode with 128-bit, 192-bit, and 256-bit keys and in GCM mode with 128-bit and 256-bit keys.
FCS_COP.1/SigGen	The TOE supports signature generation and verification for RSA (2048, 3072, and 4096 bits) and ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4. RSA and ECDSA are used in IKE authentication. ECDSA is used to verify the signature on firmware updates.
FCS_COP.1/Hash	The TOE provides cryptographic hashing services for key generation using SHA-256 as specified in NIST SP 800-90 DRBG. SHA-1, SHA-256, and SHA-384 are used in support of TLS. SHA-256, SHA-384, and SHA-512 are used in support of IPsec. SHA-256 is used with ECDSA for the verification of firmware.
FCS_COP.1/KeyedHash	The TOE implements HMAC message authentication. HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 are supported with cryptographic key sizes of 160, 256, 384, and 512 bits and message digest sizes of 160, 256, 384, and 512 bits. HMAC-SHA-1 and HMAC-SHA-256 use a block size of 512-bits. HMAC-SHA-384 and HMAC-SHA-512 use a block size of 1024 bits.
FCS_HTTPS_EXT.1	The TLS Server protocol is implemented in support of the HTTPS connection to the administrative interface. The TLS implementation is described by FCS_TLSS_EXT.1. The TOE is always the receiver of HTTPS connections. The TOE's HTTPS protocol complies with RFC 2818 by implementing all the "MUST", "REQUIRED", and "SHOULD" statements. The TOE does not implement any "MUST NOT" or "SHOULD NOT" statements. The TOE initiates an exchange of closure alerts before closing a connection. The TOE does not implement an "incomplete close".
FCS_IPSEC_EXT.1	The TOE implements IPsec in accordance with RFC 4301. The TOE Administrator implements an IPsec policy to encrypt data between the TOE and the audit server. In general, an IPsec policy can be established to encrypt data (PROTECT). If traffic not belonging to the protected interface or subnet is found on this interface, the traffic will bypass encryption and be routed to the destination in plaintext (BYPASS). If plaintext traffic is received on a protected interface or subnet, the traffic is discarded and deleted (DISCARD). This section describes IPsec rule configuration and processing. Note that when the TOE device is placed in NDPP mode, only the Protection Profile allowed algorithms are supported and visible to the administrator. NDPP mode is a configuration setting. IPsec VPN traffic is secured in two stages: • Authentication: The first phase establishes the authenticity of the sender and receiver of the traffic using an exchange of the public

Requirement	TSS Description

	key portion of a public-private key pair. This phase must be successful before the VPN tunnel can be established.
	• Encryption: The traffic in the VPN tunnel is encrypted using AES.
	
	The exchange of information to authenticate the members of the VPN and encrypt/decrypt the data uses the Internet Key Exchange (IKE) protocol for exchanging authentication information (keys) and establishing the VPN tunnel. The TOE supports IKE version 2.
	
	IKEv2 is the default proposal type for new VPN policies. Child SAs can be created, modified, and deleted independently at any time during the life of the VPN tunnel.
	
	IKEv2 initializes a VPN tunnel with a pair of message exchanges (two message/response pairs).
	
	• Initialize communication: The first pair of messages (IKE_SA_INIT) negotiate cryptographic algorithms, exchange nonces (random values generated and sent to guard against repeated messages) and perform a public key exchange.
	  ○ Initiator sends a list of supported cryptographic algorithms, public keys, and nonce.
	  ○ Responder sends the selected cryptographic algorithm, the public key, a nonce, and an authentication request.
	• Authenticate: The second pair of messages (IKE_AUTH) authenticate the previous messages, exchange identities and certificates, and establish the first CHILD_SA. Parts of these messages are encrypted, and integrity protected with keys established through the IKE_SA_INIT exchange, so the identities are hidden from eavesdroppers and all fields in all the messages are authenticated.
	  ○ Initiator sends identity proof, such as a shared secret or a certificate, and a request to establish a child SA.
	  ○ Responder sends the matching identity proof and completes negotiation of a child SA.
	
	This exchange consists of a single request/response pair. It may be initiated by either end of the SA after the initial exchanges are completed.
	
	All messages following the initial exchange are cryptographically protected using the cryptographic algorithms and keys negotiated in the first two messages of the IKE exchange.
	
	Either endpoint can initiate a CREATE_CHILD_SA exchange, so in this section the term “initiator” refers to the endpoint initiating this exchange.
	
	○ The Initiator sends a child SA offer and, if the data is to be encrypted, the encryption method and the public key.
	○ The Responder sends the accepted child SA offer and, a public key.
	
	The TOE administrative interface provides a VPN Policies page on which the policies applicable to a particular VPN can be displayed. This page has four tabs (General, Proposals, Advanced, Client) to enter the appropriate

Requirement	TSS Description
	rules. The rules for processing both inbound and outbound packets are determined by these policies. Site to Site Policies apply when the device acts as a remote client headend. In this case, the IPsec Primary Gateway Name or Address is set to 0.0.0.0. On the Network tab, the Administrator selects 'Use IKEv2 IP pool'. The pool is created with the addresses that are to be provided to the remote clients. Any required third-party certificates would have to be loaded on the VPN clients. The TOE can be only operated in Tunnel mode in the evaluated configuration. This is a default setting and cannot be changed when using IKEv2. AES-CBC-128, AES-CBC-192, and AES-CBC-256 are supported for ESP. The HMAC implementation conforms to HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512. The IKE payload is encrypted using AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128, or AES-GCM-256. The IKEv2 SA lifetime is selected in the SPD and can be set to be between 120 and 86400 seconds (24 hours). The IKEv2 Child SA lifetime is selected in the SPD and can also be set to be between 120 and 28800 seconds (8 hours). The TOE supports Group 14, 256-bit Random ECP Group (Group 19), 384-bit Random ECP Group (Group 20), and 521-bit Random ECP Group (Group 21). The DRBG described in FCS_RBG_EXT.1 is used to generate each nonce for DH groups 14, 19, 20, and 21 for IKEv2, having possible lengths of 224, 256, 384, and 512 bit, corresponding to each of the supported DH group. The TOE supports SHA-256, SHA-384, and SHA-512 as the hash in the PRF. The size of the nonce is 128-256 bits (half of the pseudorandom function with a minimum of 128 bits). The DH Group selection can be made in the VPN Policy page. The symmetric algorithms supported for IKEv2 IKE_SA uses the same or greater key length as the symmetric algorithms used to protect IKEv2 CHILD_SA. The available options ensure that the IKEv2 IKE_SA symmetric algorithm key length is equal to or greater than the IKEv2 CHILD_SA symmetric algorithm key length. Peer authentication is performed using third-party RSA or ECDSA certificates that conform to RFC 4945. Reference identifiers are supported for SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN, and Distinguished Name (DN). SAN takes precedence over CN. The format of any Subject Distinguished Name is determined by the issuing Certification Authority. Common fields are Country (C=), Organization (O=), Organizational Unit (OU=), Common Name (CN=), Locality (L=), and vary with the issuing Certification Authority. The actual Subject Distinguished Name field in an X.509 Certificate is a binary object which is converted to a string and compared with the expected string.

Requirement	TSS Description
FCS_RBG_EXT.1	The TOE implements a DRBG in accordance with ISO/IEC 18031:2011 using Hash_DRBG. The DRGB is seeded using 880-bits from a third-party entropy source provided by the Cavium Octeon hardware on the hardware appliances. The third-party entropy source is assumed to have at least .5 bits of entropy per byte, so the DRBG is seeded with at least 256 bits of entropy. The entropy source is discussed in more detail in the Entropy documentation.
FCS_TLSS_EXT.1	The TOE operates as a TLS server for the web GUI trusted path. The server only allows TLS protocol version 1.2 and rejects all other protocol version, including SSL 2.0, SSL 3.0, TLS 1.0 and TLS 1.1 and any other unknown TLS version string supplied. The TLS server is restricted to the following ciphersuites: • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 The ciphersuites are not configurable. The TLS server negotiates ciphersuites that include RSA and ECDSA key agreement schemes. For RSA key agreement schemes, key agreement parameters are restricted to 2048-bits, 3072-bits, and 4096-bits. For ECDSA key agreement schemes, the key agreement parameters are restricted to secp256r1, secp384r1, and secp521r1 curves. The TLS server supports session resumption based on session tickets and session IDs. Session tickets adhere to the structural format provided in section 4 of RFC 5077. Session IDs adhere to the structural format provided in RFC 5246. Session tickets and session IDs are encrypted according to the TLS negotiated symmetric encryption algorithm derived from the TLS handshake. Session resumption and establishment require session tickets and session IDs. The TOE-generated session IDs are used for session resumption and establishment in the Server Hello message in the TLS handshake. When session tickets are used, the TOE generates session tickets after the initial handshake.

Requirement	TSS Description
FDP_RIP.2	The TOE ensures that no data is reused with processing network packets. Once packets have been sent from the TOE, the memory buffers are allocated to the buffer pool. When memory is returned to the buffer pool, the memory is overwritten with pseudo random data. The cleared memory can then be reallocated in support of the next request.
FIA_AFL.1	The SonicWall appliance can be configured to lockout an administrator on the remote administration interface if incorrect login credentials are provided. This is configured using the Enable Administrator/User Lockout features. The number of failed attempts per minute before lockout can be set. The Lockout period, which is the time that must elapse before the user is allowed to attempt to login again, can also be set. If a user enters the configured number of incorrect login credentials, the user is blocked from submitting additional credentials until the lockout period has expired. However, the local console is not subjected to a lockout.
FIA_PMG_EXT.1 FIA_UIA_EXT.1 FIA_UAU_EXT.2 FIA_UAU.7	The SonicOS Management UI is the application used to manage the TOE devices. It is protected by HTTPS. A directly connected serial console provides a local text-based interface to manage the TOE. A management session is established with the appliance. Then, a login screen displaying the administrator-configured warning banner is presented to users. Once the warning banner is accepted, in the user authentication page, there are links to Sonicwall's knowledge-base web pages that are available for the public which the users can access before the identification and authentication. The user must be identified and authenticated prior to being granted access to any security functionality. In the evaluated configuration, only the local authentication mechanism (where username and password are stored within the device) is supported. The logon process for the SonicOS Management UI and console both require that the user enter the username and password on the logon screen. Passwords are obscured with dots to prevent an unauthorized individual from inadvertently viewing the password. The TOE hashes the user entered password and compares it to the stored hash for the associated username. The authentication is considered successful and access is granted if the hashes match. If unsuccessful, the logon screen will be displayed. No security functionality is available prior to login other than viewing the previously mentioned warning banner (except for links to Sonicwall's public KB web pages). Passwords must meet the rules set by the administrator. These rules are governed by the requirements described in FIA_PMG_EXT.1. Minimum password lengths are configurable for 15 to 99 characters. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: ["!", "@", "#", "$", "%", "^", "&", "*", "(", ")"]
FIA_X509_EXT.1/Rev FIA_X509_EXT.3	The validity of certificates is checked on certificate import and prior to usage of the public key within the certificate. Certificate validation includes checks of: • the certificate validity dates

Requirement	TSS Description
	• the validation path, ensuring that the certificate path terminates with a trusted CA certificate • basicConstraints, ensuring the presence of the basicConstraints extension • revocation status, using OCSP • extendedKeyUsage properties, when the certificate is used for OCSP The certificate path validation algorithm is implemented as described in RFC 5280. The certificate path is also validated when a certificate is imported. This validation includes a check of the certificate chain, and the keys of each of the certificates in the chain. The validity period of the certificate is also checked at this time. When a certificate is used for secure channels, an OCSP server is contacted to verify that the certificate is still valid. If the validity of a certificate that is used for secure channels cannot be verified, the system rejects the certificate and drops the connection for secure channels. For the Certificate Signing Request, a SAN is mandatory and CN is not required. The SAN may be an IP address, Domain Name, or an email address.
FIA_X509_EXT.2	Certificates are used for IPsec, TLS (HTTPS). Certificates used for IPsec are assigned a name when imported and are selected by name when the parameters are selected for an IPsec Security Policy. The certificate used for TLS/HTTPS is called the 'HTTPS Management Certificate' and is created for that purpose on the TOE device. Certificates are supplied back to the clients (the TOE only acts as the receiver of connections) and client certificates are neither required nor validated. If the validity of a certificate cannot be verified, the system rejects the certificate and drops the connection.
FMT_MOF.1/ManualUpdate FMT_MOF.1/Services FMT_MTD.1/CryptoKeys FMT_MTD.1/CoreData FMT_SMF.1 FMT_SMR.2 FMT_SMF.1/VPN FMT_SMF.1/IPS FMT_SMF.1/FFW	The TOE security functions are managed locally and remotely through the web-based management interface and restricted to authorized users assigned the Security Administrator role. Security Administrators must authenticate with the TOE prior to accessing any of the administrative functions. Manual updates to the TOE may only be performed by Security Administrators. No management of TSF data may be performed through any interface prior to login. Only administrators may login to the administrative interface, ensuring that access to TSF data is disallowed for non-administrative users. Specifically, the security administrator is able to perform the following functions: • Administer the TOE locally and remotely; • Configure the access banner; • Configure session inactivity time before session termination or locking; • Manually update the TOE; • Configure the authentication failure parameters;

Requirement	TSS Description
	• Configure the cryptographic functionality; • Generate and delete cryptographic keys (generate and delete the cryptographic keys associated with CSRs); • Configure the IPsec functionality; • Ability to modify (enable/disable) transmission of audit records to an external audit server; • Ability to set the time; • Import and delete X509 Certificates; • Configure firewall rules; • Configure packet filtering rules and associated parameters; • Enable and disable signatures applied to sensor interfaces; • Modify parameters for IPS/IDS; • Import IPS signature databases and custom create IPS signatures; • Configure anomaly detection, time-based detection/prevention, thresholds, known-good & known-bad IP lists, and actions. Rules for VPN traffic are configured through the Firewall Access Rules. The Administrator navigates to Firewall > Access Rules and selects the 'Matrix' checkbox. Under 'Zones', the Administrator can select VPN to LAN, WAN or VPN and then configures the rules. This will configure rules specifically for the VPN traffic. Firewall Access Rules for non-VPN traffic are configured using the same method by selecting the appropriate zones. Administrators can configure the IPS data analysis by selecting signatures from a pre-loaded list or by creating custom signatures. Custom Signatures are created using a combination of Application and Access rules. If a signature calls for matching L3/L4 header content, the Packet Dissection Filter can be used in conjunction with the rules. If the signature calls for application layer header/data matching, the application rules can be created with custom policy and match objects to match the desired offset in the application layer header or payload. The IPS data analysis configuration options provide the ability to deploy selections globally to either all WAN or all LAN interfaces. The access rule policies can be configured to Allow, Deny, and Discard undesired traffic. All the management functions can be performed via Web GUI and local console by security administrators.
FPT_APW_EXT.1	The TSF protects the administrator passwords used to access the device. Passwords and other sensitive data in the configuration file is protected with AES-256 hash. The user interface does not support viewing passwords.
FPT_SKP_EXT.1	The TSF does not include any function that allows symmetric keys or private keys to be displayed or exported. The use of shared secrets is not supported in the evaluated configuration. Keys may only be accessed for the purposes of their assigned security functionality.
FPT_STM_EXT.1	The TOE provides reliable time stamps that are used for audit records. The System > Time page of the web management GUI may be used to configure the time and date settings. In the evaluated configuration, time is set manually. This may be configured by deselecting 'Set time automatically using NTP and populating the appropriate values for daylight

Requirement	TSS Description
	savings time adjustments and time format. Only authorized administrators have the required privilege to set the time. Time is maintained by the system clock, which is implemented in the TOE hardware and software. Changes to the time are audited. Therefore, the time services provided are considered to be reliable. Authorized administrators may make changes to the time using the GUI.

FPT_TST_EXT.1
 	
FPT_TST_EXT.3
	The TOE performs a power on self-test on each device when it is powered on. The following hardware tests are performed:
	
	• CPU Test - This includes tests and set-up of the following:
	  ○ MMU
	  ○ Memory
	  ○ I/O ports
	  ○ Interrupts
	  ○ Timers
	• RAM Test - A memory test is performed.
	
	Following these hardware tests, the TSF performs self-tests on the cryptographic module. The following cryptographic algorithm self-tests are performed by the cryptographic module at power-up:
	
	• Firmware integrity test
	• AES-CBC/AES-GCM Encrypt and Decrypt Known Answer Tests
	• SHA-1, -256, -384, -512 Known Answer Tests
	• HMAC-SHA-1, -256, -512 Known Answer Tests
	• DSA Signature Verification Pairwise Consistency Test
	• RSA Sign and Verify Known Answer Tests
	• DH Pairwise Consistency Test
	• DRBG Known Answer Test
	• ECDSA Known Answer Test
	• ECDSA Signature and Verification Known Answer Tests
	
	For the memory test, 32K bytes of memory are tested in two steps. First, 1 or 0 is written to memory and read to verify. After that, a specific value will be written to the memory and be compared.
	
	For the physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image. For virtual appliances, the RSA signed SHA-512 hash of the image is verified.
	
	If any of the tests fail, the cryptographic module enters the error state. No security services are provided in the error state. Upon successful completion of the Diagnostic Phase, the cryptographic module enters the Command and Traffic Processing State. Security services are only provided in the Command and Traffic Processing State. No VPN tunnels are started until all tests are successfully completed. This effectively inhibits the data output interface. When all tests are completed successfully, the Test Light Emitting Diode (LED) is turned off.
	
	The SonicWall device is essentially a Finite State Machine that is synonymous with the cryptographic module. Therefore, the cryptographic

Requirement	TSS Description
	module self-tests are entirely sufficient to demonstrate the correct operation of the TOE.
FPT_TUD_EXT.1	TSF software can be updated through the web interface using the System > Settings page. This page displays the current firmware image version. To update the firmware, the administrator must first download the firmware update from SonicWall and save it to an accessible location. The administrator then selects the 'Upload New Firmware' button and 'Browse' to navigate to the firmware on the local drive. Once selected, the administrator selects 'Upload'. The digital signature on the firmware is automatically verified using the SonicWall public key. This key is appended to each firmware image made available to customers and is used to verify the new firmware. When a new firmware image is loaded on the claimed TZ, NSa, and NSSP physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image. When a new image is loaded on a virtual appliance, the cryptographic module verifies the RSA signed SHA-256 hash of the image. If the signature verification succeeds, the firmware is automatically installed. If the signature verification fails, the firmware is not loaded and an error appears. Firmware can be uploaded, but not activated. The new firmware will not be activated until the administrator boots the device with the new firmware by selecting the new firmware and 'Boot'. The version of firmware running may be queried through the TOE UI. The version of the most recently installed firmware may also be queried through the TOE UI.
FPT_FLS.1/SelfTest	An integrity check of the TSF executable image is run when the image is loaded. A Continuous Random Number Generator Test (CRNGT) is performed on the output of the entropy source prior to seeding the FIPS Approved DRBG to provide health testing of the noise source. Power-on Self-tests are run during boot up. If any of these self-tests fail, the device enters an error state. At this point, a user must power the device down and restart to attempt to resolve the error.
FTA_SSL_EXT.1 FTA_SSL.3 FTA_SSL.4 FTA_TAB.1	All access to the TOE takes place through the web-based management interface over HTTPS or the local serial console. The web-based management interface can be accessed using the GUI (Note that the Admin Guides refers to the GUI as the MGMT interface). Inactive local and remote sessions to the TOE are automatically terminated after a Security Administrator-configurable time interval between 1 and 9999 minutes. By default, the TOE terminates a session after five minutes of inactivity. In addition, administrators are provided with the capability to terminate their own local or remote session using the instructions provided in the administrative guides. All users, both local and remote, are presented with a Security Administrator-configured advisory notice and consent warning prior to TOE login.
FTP_ITC.1 FTP_ITC.1/VPN FTP_TRP.1/Admin	IPsec VPN tunnels are used to provide a trusted communication channel between the TOE and the external audit server and to support VPN communications. The exchange of information to authenticate the members of the VPN and encrypt/decrypt the data uses the Internet Key

Requirement	TSS Description
	Exchange (IKE) protocol for exchanging authentication information (keys) and establishing the VPN tunnel. The TOE supports IKE version 2 in protecting these communications from disclosure and detecting modification. HTTPS is used to provide a trusted path for communications between the TOE and the administrative interface. The TOE supports TLS 1.2 to protect these communications from disclosure and detect modification. All other protocol requests will be denied. RSA with 2048-bits, 3072-bits, and 4096-bits keys is used in the supported TLS ciphersuites.
FFW_RUL_EXT.1 FPF_RUL_EXT.1	Packets are received by the SonicWall device on one of three Ethernet links: the LAN, WAN, or optional DMZ link. A flag called gStartupTrulyComplete is set after firewall bootup to identify when the network stack and the policy are ready to process packets. Before this flag is set to TRUE, firewall initializes the interfaces but set the interfaces to DOWN. Only after gStartupTrulyComplete is set to TRUE, TOE enables the interfaces. Once the interfaces are enabled, the packets are analyzed in the communications stack at a level that is best described as above the Ethernet driver, but below the networking stack. Transport-and application-layer data is also examined. This higher-level data is used to provide the stateful inspection security. During this analysis, packets are modified, dropped, passed up to the networking stack, or rewritten directly to another Ethernet link, as appropriate. The analysis is based on a set of rules entered by the firewall administrator which can be tied to the LAN, WAN, or optional DMZ links, despite the interfaces being standalone or grouped via link aggregation. The SonicWall device acts as a single component. If the component fails, processing ceases and all traffic is stopped. SonicWall interacts with the Ethernet drivers, and also with the networking stack. An incoming packet will initially be read by the Ethernet driver. At this point, the device does one of three things: • Drop the packet. It will do this based on the security policy configured by the administrator • Rewrite the packet, which may be modified, to another Ethernet link • Pass the packet up to the stack Conceptually, the stack exists on the LAN link of the SonicWall. If the stack tries to communicate with the DMZ or Internet, then the device will provide network address translation. When an Ethernet packet is received on a given link, Address Resolution Protocol (ARP) and Point-to-Point Protocol over Ethernet (PPPoE) packets are first vectored off to their respective handlers. IP packets are sent through a complicated series of code modules that analyze them, modify them, forward them, or drop them, as appropriate. The path of a packet through these code modules is described here.

Requirement	TSS Description

	First, raw fields of the packet buffer are analyzed and unpacked into a machine-aligned structure. This is done for optimization; endian conversion and alignment shifting only happens once.
	
	Next, the packet goes through a sequence of stateless analysis. That is, the packet is analyzed based solely on the contents of the packet, not taking the connection into account.
	
	• IPSec packets are vectored to the IPSec handling code. This essentially encapsulates and encrypts (or unencapsulates and decrypts) the packet. Conceptually, the IPSec tunnel terminates on the inside of the firewall, so packets are encrypted before passing through the firewalling, content filtering, and other code. Conversely, incoming traffic is decrypted and then written to the LAN without filtration.
	• Stateless Attack Prevention analysis is performed. This consists of stateless checks for malformed and fragmented packets, smurf amplifiers, Layer 4 Denial of Service (LAND) attacks, etc. The analysis code may decide to drop the packet and create a log message.
	• Packets addressed to the firewall itself may be vectored off at this point. For instance, TCP packets directed to the management interface may be passed up the stack. Packets may be sent directly to code modules without depending on the stack. For example, UDP packets may be directed to the DHCP server or client.
	• DNS packets may be intercepted in order to support domain-name access to the firewall without configuration of a DNS server, and also to foil a bug with IE4 involving reverse-DNS lookups for java applets.
	• Packets may be bounced off the LAN interface if they have been routed improperly; ICMP redirect packets are sent in an attempt to rectify the problem.
	
	Next, the packet goes through a sequence of stateful analysis.
	
	• A connection cache lookup takes place. If a cache entry isn't found, one is added (even if this packet will be dropped).
	• Incoming packets must be NAT-remapped during this cache lookup process in order to find them properly. From this point on, the destination IP and port information will be remapped to internal, private values.
	• Stateful attack prevention is performed.
	    o SYN floods are detected, and any suspicious connections are reset. Technically, this step happens BEFORE the connection cache lookup. This is because SYN flood prevention uses a different cache than the main connection cache. This is mostly for historical reasons; it may be changed in the future. (In versions 1.x, there was no firewalling of the DMZ; only attack prevention).
	    o IP Spoof checking is simply a sanity check of the source and destination IP addresses against the static routing

Requirement	TSS Description
	information in the box. This could be done statelessly, however, there is a significant speed advantage when cached routing information is used. ○ TCP sequence numbers are offset by a random value for every distinct TCP connection. • Antivirus policing may redirect a web query to the Virus Update website if the client's antivirus software is out of date. • User-based authentication tables are checked; these may override packet filtration or content filtration. • Packet filtering rules are checked. If the packet matches an 'ALLOW' access rule, the connection cache is created. If the packet matches a 'DENY' rule, or there is no matched 'ALLOW' rule, the packet does not proceed. • Stateful inspection takes place. This is a set of application-specific code modules that examine application-layer packet contents in order to add 'anticipated' cache elements on the fly. In other words, a cache element will be added for a connection that would normally violate the packet filtering rules, such as an incoming FTP data connection. Since the cache element already exists by the time the first incoming SYN packet arrives, it will not be rejected by the packet filtration. • Content filtration takes place. This is primarily for Web traffic, although some filtration can be done on other protocols. Note that it is not sufficient to identify traffic using TCP port 80, since some web sites use non-standard ports. The SonicWall device checks for a 'GET /' command in the application-layer data. ○ Cybernot list ○ Trusted and forbidden domains ○ ActiveX, Java, and Cookie blocking ○ Keyword scanning ○ Proxy servers blocking • License enforcement takes place. For instance, connections from the eleventh IP address on the LAN of a 10-user SOHO box will be rejected. • Outgoing packets are NAT-remapped. From this point on, the source IP and port information will be set to external, valid Internet values. (That is, unless the WAN port is on its own private network). • Proxy redirection may take place, if the firewall is configured to send all web traffic through an external proxy such as a web cache. This is done by prepending some data to pieces of the web command, and then changing the destination IP address to match the proxy server rather than the actual web server. Finally, the packet is written back to the network. The Ethernet link used to write the packet (LAN, WAN, or DMZ) is determined by the static

Requirement	TSS Description

	routing information stored in the firewall's configuration. After the packet is written out, some cleanup takes place, and then the packet is done.
	
	If any component fails, packets will not be accepted into the connection cache, and will therefore not be allowed to flow through the device.
	
	The following RFCs are supported:
	• RFC 792 (ICMPv4)
	  ▪ *Type; and*
	  ▪ *Code*
	• RFC 4443 (ICMPv6)
	  ▪ *Type; and*
	  ▪ *Code*
	• RFC 791 (IPv4)
	  ▪ *Source address;*
	  ▪ *Destination Address; and*
	  ▪ *Transport Layer Protocol*
	• RFC 8200 (IPv6)
	  ▪ *Source address;*
	  ▪ *Destination Address;*
	  ▪ *Transport Layer Protocol*
	• RFC 793 (TCP)
	  ▪ *Source Port; and*
	  ▪ *Destination Port*
	• RFC 768 (UDP)
	  ▪ *Source Port; and*
	  ▪ *Destination Port*
	
	Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.
	
	The Stateful packet filtering policy consists of the following rules and attributes.
	• Action: (Allow/Deny/Discard)
	  ○ Configure to permit or drop the packet
	• From: (Zone/Interface)
	  ○ Packet ingress point
	• To: (Zone/Interface)
	  ○ Packet egress point
	• Source Port: (Services Object)
	  ○ The protocol and the source port of the packet
	• Services: (Services Object)

Requirement	TSS Description
	◦ The protocol and the destination port of the packet • Source: (Host/Range/Network) • Source IP: The source IP of the packet • Destination: (Host/Range/Network) • Destination IP: the Destination IP of the packet • Enable Logging (Checkbox) • Log the action when it is taking place • TCP Connection Inactivity Timeout (minutes) • UDP Connection Inactivity Timeout (seconds) The attributes are all configurable for ICMPv4, ICMPv6, IPv4, IPv6, TCP and UDP policies. Logging can be configured for each access rule. The source and destination address are configurable for each access rule. The supported header fields for IPv4, IPv6, TCP, UDP, ICMPv4 and ICMPv6 are listed below in **IPS_SBD_EXT.1**. Stateful session handling is supported for TCP and UDP. Source and destination addresses, and source and destination ports are used together to recognize TCP flow in support of stateful session handling. Sequence numbers are used to ensure that the received data falls within the window defined for the protocol. Flags are used to track the connection against the defined TCP State Machine states: • Listen State: Only a TCP packet with just the SYN flag is considered valid. • Syn-Sent State: ◦ ACK number (if present) must be valid. ◦ RST packet (with a valid TCP ACK number) is valid. ◦ FIN packet (which does not have the SYN bit set) is also considered valid. • Syn-Received, Established, Fin-Sent, and Fin-Acked States: ◦ SEQ number must be within the TCP window for the destination or be that for Keep-Alive packet. ◦ RST packet (with a valid TCP SEQ number) is valid. ◦ ACK number must also be present and valid in this state. ◦ A SYN seen in this state will cause the TCP connection to be closed. • Close-Wait State: ◦ A SYN is valid (to re-open the same TCP connection). ◦ Any other packet which is also valid in the previous state is acceptable. For UDP, source and destination addresses, and source and destination ports are used together to be checked to match with an access rule. Following a UDP request, the TOE will accept return packets for a configurable period of time. This is generally in the order of several seconds and is configurable as the UDP Timeout in the applicable access rule. Stateful sessions are removed when complete, or when the timeout is triggered.

<table>
<tr><th>Requirement</th><th>TSS Description</th></tr>
<tr><td></td><td>
For TCP connection completion, the connection is closed in one of two ways:

Syn-Sent State
A validated RST will cause the action of the TCP connection to be closed.
Syn-Received, Established, Fin-Sent, Fin-Acked, and Close-Wait States
A validated RST will cause the action of the TCP connection to be closed.
Acknowledged TCP FINs will cause the action of the TCP connection to be closed.

Session removal becomes effective immediately after Connection cache is removed.
Each packet flow through the TOE triggers a timestamp update to its connection cache. The TOE checks this timestamp, and if the connection cache timeout has been reached, the session is removed.
The TOE will automatically drop and log the event when the following is found:

A packet is found to be an invalid fragment. A fragment is determined to be invalid if it cannot be combined with other fragments to form a packet. The offset may be incorrect, or it may be considered to be too small
A fragmented packet cannot be completely re-assembled
A packet with a source address that is defined as being on a broadcast network
A packet with a source address that is defined as being on a multicast network
A packet with a source address that is defined as being a loopback address
A packet with a source or destination address that is defined as unspecified or reserved for future use as specified in RFC 5735 for IPv4
A packet with a source or destination address that is defined as an unspecified address or an address reserved for future definition and use as specified in RFC 3513 for IPv6.
A packet with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified
Packets where the source address is equal to the address of the network interface where the network packet was received
Packets where the source or destination address of the network packet is a link-local address
Packets where the source address is not identified by the routing table as a network associated with the network interface the packet was received

The algorithm applied to incoming packets performs the following actions:
</td></tr>
</table>

Requirement	TSS Description

	• In the evaluated configuration, the default action is to DENY a packet. The TOE checks the incoming packet against all of the access rules. If the packet does not match any access rule and does not belong to an approved established connection, then the default action is to DENY the packet.
	• The TOE performs a Connection cache lookup
	  ○ each connection cache represents an established session
	  ○ For incoming packets, srcIP, dstIP, srcPort, dstPort, IPType are used together as a hash index to find the matched connection cache
	  ○ An access rule check is performed if the connection cache lookup fails
	• The TOE performs an access rule check only if the connection cache lookup fails. The following rules are applied in an access rule check:
	  ○ Access rules are ordered by Priority. The rule with higher Priority will be applied
	  ○ For incoming packets, srcZone, dstZone, srcIP, dstIP, srcPort, dstPort, IP type are used together as a hash index to find the matching access rule
	  ○ If an incoming packet matches an access rule with the ALLOW action, a new connection cache is added. Otherwise the packet is dropped
	
	In the evaluated configuration, the default action is to DENY a packet if the packet does not match any of the access rules. However, this does not apply for dynamic protocol traffic.
	
	The TOE does not allow configuring conflicting rules. To identify conflicting rules, the TOE will validate the action (allow or deny), source and destination IP addresses, protocols, services, and source and destination interfaces/zones attributes of the rules at the time of creation. If any conflict is detected, the rule is not allowed to be created and an error will be displayed indicating that a conflicting rule is present.
	
	Dynamic protocols include ftp, tftp, pptp, and oracle. These protocols are similar to ftp in that they use multiple TCP connections. The first connection is the control connection. A particular command, specified by the protocol, opens the one or more additional data connections. The TOE inspects the control connection to find the target commands and adds the new connection cache appropriate to allow the network traffic.
	
	The TOE tracks and maintains information relating to the number of half-open TCP connections as follows:
	
	• There is an administratively defined limit for half-open TCP connections based on:
	  ○ TCP Handshake Timeout (seconds)
	  ○ Maximum Half Open TCP Connections
	• There is a TCP Handshake Timeout (seconds)
	  ○ Each half-open TCP connection is removed if the handshake is not complete by the time this timeout is reached

Requirement	TSS Description
	• There is a maximum number of allowable Half Open TCP Connections ○ A global counter is used by the TOE to track the number of all half-open TCP connections. When this number reaches the value of Maximum Half Open TCP Connections, new incoming TCP connections are dropped.
IPS_ABD_EXT.1	The TOE supports baseline and anomaly-based traffic based on time of day. If traffic is received outside of the permitted time of day, the TOE may block or drop the flow of traffic. This rule can be applied to any WAN or LAN network interface. Subsequently, if traffic is received within the permitted time of day, the TOE may allow the traffic to flow.
IPS_IPB_EXT.1	IPS policies are configured by defining a known good list ('included') and a known bad list ('excluded') of IP addresses for each IPS Signature. Known-good IP addresses are allowed to pass through the TOE to their destination. Known bad IP addresses are blocked from accessing the network. IP addresses can be defined by a single IP or by a range of IP addresses. Only authorized users assigned the Security Administrator role can access and configure the IPS policies.
IPS_NTA_EXT.1	The TOE analyzes traffic based on IP address, port, and interface. The traffic is first analyzed against the anomaly-based rules and then against the signature-based rules. This policy hierarchy order is not configurable. All traffic is allowed until configured by a Security Administrator. Depending on the model, the TOE supports a number of WAN and LAN interfaces capable of implementing IPS policies while in inline mode and promiscuous mode. All policies including signature-based, baseline, and anomaly-based are deployed globally across all WAN and LAN interfaces. Each instance of the TOE also supports a management interface (MGMT port) used only for the web-based administration of the TOE. The MGMT port is distinctly labeled on each device. The TOE supports the following protocols, which have been compliance tested by the product QA team for assurance: • IPv4 • IPv6 • ICMPv4 • ICMPv6 • TCP • UDP
IPS_SBD_EXT.1	Signature rules are comprised of the following settings: • Interface (WAN/LAN) • Source Port • Service • Destination • Included/Excluded Users • Schedule

Requirement	TSS Description

	Administrators can download a pre-determined list of signatures from SonicWall and/or manually create custom signatures to be applied to sensor interfaces and inline interfaces. The only difference between the inline and sensor (promiscuous mode) interfaces is the intrusion prevention vs intrusion detection. The way the IPS signature rules are applied is the same for both interface modes. Inline mode interfaces can detect and prevent traffic based on IPS rules while the sensor mode can only detect. For management mode interfaces, IPS signature rules cannot be applied. Management mode interface is used as an out-of-bound interface dedicated to being used in a dedicated management network.
	
	By analyzing the header-based signature traffic, the TOE is able to detect and prevent the following types of attacks:
	
	• IP Attacks
	    ○ IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)
	    ○ IP source address equal to the IP destination (Land attack)
	• ICMP Attacks
	    ○ Fragmented ICMP Traffic (e.g. Nuke attack)
	    ○ Large ICMP Traffic (Ping of Death attack)
	• TCP Attacks
	    ○ TCP NULL flags
	    ○ TCP SYN+FIN flags
	    ○ TCP FIN only flags
	    ○ TCP SYN+RST flags
	• UDP Attacks
	    ○ UDP Bomb Attack
	    ○ UDP Chargen DoS Attack
	• Flooding a host (DoS attack)
	    ○ ICMP Flooding (Smurf attack, and ping flood)
	    ○ TCP flooding (e.g. SYN flood)
	• Flooding a network (DoS attack)
	• TCP Attacks
	    ○ IP protocol scanning
	    ○ TCP port scanning
	    ○ UDP port scanning
	    ○ ICMP scanning
	
	When a packet is received by the TOE, the header and payload data elements are analyzed and compared to the list of signatures to identify any policy violations. Reactions to all signature policy violations can be set to either Detection or Prevention. If Detection is enabled, the TOE identifies the policy violation, logs the instance, and allows the traffic to flow through. If Prevention is enabled, the TOE reacts by identifying the violation, logging the instance, and blocking or dropping the traffic. For TCP sequence number errors, the TOE can remap the sequence number and forward the traffic to its destination.
	
	The TOE supports string-based detection signatures by inspecting the payload data elements. String-based pattern matching with the data elements of the following protocols are also supported:

Requirement	TSS Description

	• ICMPv4 data: characters beyond the first 4 bytes of the ICMP header.
	• ICMPv6 data: characters beyond the first 4 bytes of the ICMP header.
	• TCP data (characters beyond the 20 byte TCP header), with support for detection of:
	  ○ FTP (file transfer) commands: help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.
	  ○ HTTP (web) commands and content: commands including GET and POST, and administrator defined strings to match URLs/URIs, and web page content.
	  ○ SMTP (email) states: start state, SMTP commands state, mail header state, mail body state, abort state.
	• UDP data: characters beyond the first 8 bytes of the UDP header;
	
	To properly detect configured strings within streams, the TOE supports stream reassembly to detect malicious payloads even if split across multiple non-fragmented packets.

6.1 CAVP Algorithm Certificate Details

Each of these cryptographic algorithms have been validated as identified in the table below.

Table 32 – Algorithm to SFR and CAVP Certificate Mapping

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
FCS_CKM.1	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	SonicOS/X 7.0.1 for TZ, NSA Series	RSA KeyGen (FIPS186-4) Moduli: 2048, 3072, 4096	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	RSA KeyGen (FIPS186-4) Moduli: 2048, 3072, 4096	A2583
		SonicOS/X 7.0 for NSv Series	RSA KeyGen (FIPS186-4) Moduli: 2048, 3072, 4096	A4982
	ECC schemes using "NIST curves" [selection: P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4	SonicOS/X 7.0.1 for TZ, NSA Series	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A2583
		SonicOS/X 7.0 for NSv Series	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A4982
	FFC Schemes using 'safe-prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526]	SonicOS/X 7.0.1 for TZ, NSA Series	Safe Primes Key Generation Safe Prime Groups: modp2048 Safe Primes Key Verification Safe Prime Groups: modp2048	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	Safe Primes Key Generation Safe Prime Groups: modp2048 Safe Primes Key Verification Safe Prime Groups: modp2048	A2583
		SonicOS/X 7.0 for NSv Series	Safe Primes Key Generation Safe Prime Groups: modp2048 Safe Primes Key Verification Safe Prime Groups: modp2048	A4982
FCS_CKM.1.1/IKE	FIPS PUB 186-4, "Digital Signature Standard (DSS),"	SonicOS/X 7.0.1 for TZ, NSA Series	RSA	A5110

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
	Appendix B.3 for RSA schemes		FIPS PUB 186-4 Key Generation (2048-bit, 3072-bit, 4096-bit)	
		SonicOS/X 7.0.1 for NSa, NSsp Series	RSA FIPS PUB 186-4 Key Generation (2048-bit, 3072-bit, 4096-bit)	A2583
		SonicOS/X 7.0 for NSv Series	RSA FIPS PUB 186-4 Key Generation (2048-bit, 3072-bit, 4096-bit)	A4982
	FIPS PUB 186-4, "Digital Signature Standard (DSS)," Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-384 and [P-256, P-521]	**SonicOS/X 7.0.1 for TZ, NSA Series**	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A2583
		SonicOS/X 7.0 for NSv Series	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A4982
	FFC Schemes using "safe-prime" groups that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526]	**SonicOS/X 7.0.1 for TZ, NSA Series**	Safe Primes Key Generation Safe Prime Groups: modp2048 Safe Primes Key Verification Safe Prime Groups: modp2048	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	Safe Primes Key Generation Safe Prime Groups: modp2048 Safe Primes Key Verification Safe Prime Groups: modp2048	A2583
		SonicOS/X 7.0 for NSv Series	Safe Primes Key Generation Safe Prime Groups: modp2048 Safe Primes Key Verification Safe Prime Groups: modp2048	A4982
FCS_CKM.2	RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017,	**SonicOS/X 7.0.1 for TZ, NSA Series** **SonicOS/X 7.0.1 for NSa, NSsp Series**	No NIST CAVP, CCTL has performed all assurance/evaluation activities and documented	N/A. This testing was performed in conjunction with FTP_TRP.1/Admin Test #1 and FTP_ITC.1

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
	"Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1"	SonicOS/X 7.0 for NSv Series	in the ETR and AAR accordingly.	Test #1 to demonstrate correct operation.
	Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"	SonicOS/X 7.0.1 for TZ, NSA Series	KAS-ECC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: P-256, P-384, P-521	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	KAS-ECC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: P-256, P-384, P-521	A2583
		SonicOS/X 7.0 for NSv Series	KAS-ECC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: P-256, P-384, P-521	A4982
	FFC Schemes using "safe-prime" groups that meet the following: 'NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment collaborative Protection Profile for Network Devices v2.2e, 23-March-2020 Page 57 of 174 Schemes Using Discrete Logarithm Cryptography" and [groups listed in RFC 3526].	SonicOS/X 7.0.1 for TZ, NSA Series	KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: modp-2048	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: modp-2048	A2583
		SonicOS/X 7.0 for NSv Series	KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: modp-2048	A4982
FCS_COP.1/ DataEncryption	AES used in [**CBC, GCM**] mode and cryptographic key sizes [**128 bits, 192 bits, 256 bits**]	SonicOS/X 7.0.1 for TZ, NSA Series	AES-CBC Direction: Decrypt, Encrypt Key Length: 128, 192, 256 AES-GCM Direction: Decrypt, Encrypt Key Length: 128, 256	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	AES-CBC Direction: Decrypt, Encrypt Key Length: 128, 192, 256 AES-GCM Direction: Decrypt, Encrypt Key Length: 128, 256	A2583
		SonicOS/X 7.0 for NSv Series	AES-CBC Direction: Decrypt, Encrypt Key Length: 128, 192, 256 AES-GCM Direction: Decrypt, Encrypt Key Length: 128, 256	A4982
FCS_COP.1/ SigGen	For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1	SonicOS/X 7.0.1 for TZ, NSA Series	RSA SigGen (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096	A5110

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
	Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3		RSA SigVer (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096	
		[SonicOS/X 7.0.1 for NSa, NSsp Series]()	RSA SigGen (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096 RSA SigVer (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096	A2583
		[SonicOS/X 7.0 for NSv Series]()	RSA SigGen (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096 RSA SigVer (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096	A4982
	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	[SonicOS/X 7.0.1 for TZ, NSA Series]()	ECDSA SigGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA SigVer (FIPS186-4) Curve: P-256, P-384, P-521	A5110
		[SonicOS/X 7.0.1 for NSa, NSsp Series]()	ECDSA SigGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA SigVer (FIPS186-4) Curve: P-256, P-384, P-521	A2583
		[SonicOS/X 7.0 for NSv Series]()	ECDSA SigGen (FIPS186-4) Curve: P-256, P-384, P-521 ECDSA SigVer (FIPS186-4) Curve: P-256, P-384, P-521	A4982
FCS_COP.1/ Hash	[**SHA-1, SHA2-256, SHA2-384, SHA2-512**] and message digest sizes [**160, 256, 384, 512**] bits	[SonicOS/X 7.0.1 for TZ, NSA Series]()	SHA-1 SHA2-256 SHA2-384 SHA2-512	A5110
		[SonicOS/X 7.0.1 for NSa, NSsp Series]()	SHA-1 SHA2-256 SHA2-384 SHA2-512	A2583
		[SonicOS/X 7.0 for NSv Series]()	SHA-1 SHA2-256 SHA2-384 SHA2-512	A4982
FCS_COP.1/ KeyedHash	[**HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512**] and cryptographic key sizes [**key size (in bits) used in HMAC**] and message digest sizes [**160, 256, 384, 512**] bits	[SonicOS/X 7.0.1 for TZ, NSA Series]()	HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512	A5110
		[SonicOS/X 7.0.1 for NSa, NSsp Series]()	HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512	A2583

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
		SonicOS/X 7.0 for NSv Series	HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512	A4982
FCS_RBG_EXT.1	**Hash_DRBG**	SonicOS/X 7.0.1 for TZ, NSA Series	Hash DRBG SHA2-256	A5110
		SonicOS/X 7.0.1 for NSa, NSsp Series	Hash DRBG SHA2-256	A2583
		SonicOS/X 7.0 for NSv Series	Hash DRBG SHA2-256	A4982

Table 33 – CAVP Certificate to Operational Environment Mapping

Operational Environment	Certificate
Marvell 88F7040	A5110
Marvell CN9130	
Intel Xeon D-2123IT	A2583
Intel Xeon D-2166NT	
Intel Xeon D-2187NT	
SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)	A4982
SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)	

6.2 Cryptographic Key Destruction

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

Table 34 – Cryptographic Key Destruction

Keys/CSPs	Purpose	Storage Location	Method of Zeroization
RSA private key used for TLS	RSA (2048 bits, 3072 bits, 4096 bits)	Stored in flash memory Held in the RAM buffer in plaintext	The key is overwritten with a block erase when deleted The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
RSA public key used for TLS	RSA (2048 bits, 3072 bits, 4096 bits)	Stored in flash memory Held in the RAM buffer in plaintext	The key is overwritten with a block erase when deleted

Keys/CSPs	Purpose	Storage Location	Method of Zeroization
			The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
AES key used for TLS	AES-128 AES-192 AES-256	Keys are not stored Held in the RAM buffer in plaintext	The key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
Key Agreement Keys used for IPsec	DH (2048 bits) ECDH (P-256, P-384, P-521)	Keys are not stored Held in the RAM buffer in plaintext	The key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
Authentication Keys used for IPsec	RSA (2048 bits) ECDSA (P-256, P-384, P-521)	Stored in flash memory Held in the RAM buffer in plaintext	The key is overwritten with a block erase when deleted The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
AES Keys used for IPsec	AES-128 AES-256	Keys are not stored Held in the RAM buffer in plaintext	The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
SonicWall Public Key used to verify firmware updates	ECDSA (P-256)	Stored in Flash Memory	The key may be overwritten by a software update
Locally stored passwords	AES-256 in configuration file.	Encryption key is Hardcoded in Flash Memory	Random IV is generated every time the configuration file is updated or imported or after a reboot.

7 Acronym Table

Acronyms should be included as an Appendix in each document.

Table 35 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
CC	Common Criteria
DTLS	Datagram Transport Layer Security
EP	Extended Package
GUI	Graphical User Interface
IP	Internet Protocol
NDcPP	Network Device Collaborative Protection Profile
NIAP	Nation Information Assurance Partnership
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
PP	Protection Profile
RSA	Rivest, Shamir & Adleman
SFR	Security Functional Requirement
SSH	Secure Shell
ST	Security Target
TOE	Target of Evaluation
TLS	Transport Layer Security
TSS	TOE Summary Specification