Dell SonicWALL, Inc.

SonicOS Enhanced v5.9.0 on NSA Series and TZ Series Appliances

Security Target

Document Version: 2.5

Prepared for:

Dell SonicWALL, Inc.
2001 Logic Avenue
San Jose, CA 95124-3452
United States of America

Phone: +1 888 557 6642
http://www.sonicwall.com

Prepared by:

Corsec Security, Inc.
13135 Lee Jackson Memorial Hwy., Suite 220
Fairfax, VA 22033
United States of America

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

Table of Contents

Table of Figures

List of Tables

1 Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the organization of the ST. The TOE is one or more SonicWALL SonicOS Enhanced v5.9.0 on NSA Series and TZ Series Appliances, and will hereafter be referred to as the TOE throughout this document. The TOE is a unified threat management (UTM) device. UTMs are consolidated threat-management devices that provide multiple security services, such as network firewall, spam filtering, anti-virus capabilities, intrusion prevention systems (IPS), and World Wide Web content filtering at the network level. SonicWALL appliances also provide virtual private network (VPN), and traffic management capabilities.

1.1 Purpose

This ST is divided into nine sections, as follows:

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

1.2 Security Target and TOE References

Table 1 below shows the ST and TOE references.

Table 1 ST and TOE References

ST Title	Dell SonicWALL, Inc. SonicOS Enhanced v5.9.0 on NSA Series and TZ Series Appliances Security Target
ST Version	Version 2.5
ST Author	Corsec Security, Inc.
ST Publication Date	6/27/2013

ST Title	Dell SonicWALL, Inc. SonicOS Enhanced v5.9.0 on NSA Series and TZ Series Appliances Security Target
TOE Reference	SonicWALL SonicOS Enhanced v5.9.0 on NSA Series and TZ Series Appliances build 93 TZ 105: 101-500356-56- , TZ 105W: 101-500368-51, TZ 205: 101-500358-59, TZ 205W: 101-500359-59, TZ 215: 101-500355-57, TZ 215W: 101-500355-55, NSA 220: 101-500347-61, NSA 220W: 101-500342-60, NSA 240: 101-500240-54, NSA 250M: 101-500343-58, NSA 250MW: 101-500326-61, NSA 2400: 101-500219-53, NSA 2400MX: 101-500270-56, NSA 3500: 101-500073-50, NSA 4500: 101-500166-50, NSA E5500: 101-500088-50, NSA E6500: 101-500165-50, NSA E7500: 101-500163-50, NSA E8500: 101-500308-57, NSA E8510: 101-500344-57, NSA E10400: 101-500337-50, NSA E10800: 101-500280-50, NSA E10200: 101-500336-50
FIPS[1] 140-2 Status	Level 2, Validated crypto modules TZ 105, TZ 105W, TZ 205, TZ 205W, TZ 215, TZ 215W Certificate No. 2190 NSA 220, NSA 220W, and NSA 240 Certificate No. 2187 NSA 250M and NSA 250MW Certificate No. 2193 NSA 2400 and NSA 2400MX Certificate No. 2186 NSA 3500 Certificate No. 2188 NSA 4500 and NSA E5500 Certificate No. 2189 NSA E6500 Certificate No. 2185 NSA E7500 Certificate No. 2104 NSA E8500 and NSA E8510 Certificate No. 2191 NSA E10200, NSA E10400, and NSA E10800 Certificate No. 2192

1.3 Product Overview

SonicWALL SonicOS Enhanced v5.9.0 on NSA Series and TZ Series Appliances is custom software running on purpose built hardware platforms that combine to form a UTM device. UTMs are network firewalls that provide additional features, such as spam filtering, anti-virus capabilities, IPS, and World Wide Web content filtering[2]. The product under evaluation consists of the SonicOS Enhanced operating system (OS) for the following appliances: TZ 105, TZ 105W, TZ 205, TZ 205W, TZ 215, TZ 215W, NSA 220, NSA 220W, NSA 240, NSA 250M, NSA 250MW, NSA 2400, NSA 2400MX, NSA 3500, NSA 4500, NSA E5500, NSA E6500, NSA E7500, NSA E8500, NSA E8510, NSA E10400, NSA E10800, and NSA E10200. The appliances include the same basic functionality, provided by SonicOS Enhanced, but vary in number of processors, size of appliance, and connections they support. These appliances provide firewall, UTM, VPN, and traffic management capabilities. The product is managed using a web-based Graphical User Interface (GUI), called the Web Management GUI, accessed through a permitted device running a supported web browser connected directly to the appliance over a network cable and communicating via hypertext transfer protocol – secure (HTTPS).

SonicOS Enhanced is a proprietary operating system designed for use on SonicWALL appliances. The appliances run only signed SonicOS firmware, and are licensed to provide a selection of features to the end user. SonicOS Enhanced provides policy-based network traffic control, UTM, and VPN services.

SonicOS Enhanced's firewall capabilities include stateful packet inspection. Stateful packet inspection keeps track of 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

[1] FIPS – Federal Information Processing Standard
[2] Please note that the spam filtering and World Wide Web content filtering functionality is not included as a part of this evaluation.

between legitimate packets and illegitimate packets for the given network deployment. Only packets adhering to the administrator-configured access rules are allowed to pass through the firewall; all others are rejected.

SonicOS Enhanced's UTM capabilities include deep-packet inspection (DPI). The optional licensed services that make up the UTM include IPS, Gateway Anti-Virus (GAV), Application Intelligence and Control, and Gateway Anti-Spyware (SPY). All UTM services employ stream-based analysis wherein traffic traversing the product is parsed and interpreted so that its content might be matched against sets of signatures to determine the acceptability of the traffic. The parsing and interpretation engines allow for the reliable handling of any application layer protocol, encoding, and type of compression. In the event a certain flow of traffic is found to match an Application List signature and meets or exceeds the configured threshold, the event is logged, and the offending flow is terminated.

SonicOS Enhanced supports VPN functionality, which provides a secure connection between two or more computers or protected networks over the public internet. It provides authentication to ensure that information is going to and from the correct parties, and protects the information from viewing or tampering en-route. SonicOS Enhanced supports the creation and management of Internet Protocol Security (IPsec) VPNs. IPsec is a suite of protocols that operate on network traffic to secure Internet Protocol (IP) communications by authenticating and encrypting packets. Cryptographic key establishment is also possible through IPsec. For this, SonicOS Enhanced supports Internet Key Exchange (IKE) version 1 and 2, which is the protocol used to set up a security association (SA) in the IPsec protocol suite. SonicOS Enhanced enables VPN policy creation to provide the configuration of multiple VPN tunnels. VPN policy definitions include the IP address of the remote gateway appliance with which the product will communicate, the IP address of the destination network, the type of encryption used for the policy, and other configuration information.

SonicOS Enhanced provides site-to-site VPN functionality. Site-to-site VPN functionality allows creation of VPN policies for connecting offices running SonicWALL security appliances, resulting in network-to-network VPN connections.

Digital certificates are also supported by SonicOS Enhanced. A digital certificate is an electronic means to verify identity by a trusted third party known as a Certification Authority (CA). SonicOS Enhanced users can obtain certificates signed and verified by a third party CA to use with an IKE VPN policy. This makes it possible for VPN users to authenticate peer devices without manually exchanging shared secrets or symmetric keys. SonicOS Enhanced interoperates with any X.509v3-compliant provider of certificates. Use of certificates for authentication, including CAC[3] cards that comply to FIPS 201 is also supported.

The product implements both physical and virtual interfaces. Physical interfaces are bound to a single port. Virtual interfaces are assigned as sub-interfaces to a physical interface, and allow the physical interface to carry traffic assigned to multiple virtual interfaces. The product allows static IP address configuration on all physical and logical network interfaces, as well as dynamic configuration of Wide Area Network (WAN) interfaces through Dynamic Host Configuration Protocol (DHCP), Point to Point Protocol over Ethernet, Point to Point Tunneling Protocol PPTP, and Layer 2 Tunneling Protocol. Additionally, interface pairs may be configured in a Layer 2 Bridge mode to enable the inspection and control of traffic between the resulting two segments without a need for logical reconfiguration of the target network.

In addition, physical interfaces may be assigned to Security Zones. Zones are optional logical groupings of one or more interfaces designed to make management of the product simpler and to allow for configuration of access rules governing inbound and outbound traffic. If there is no interface, traffic cannot access the zone or exit the zone. Zones allow the administrator to group similar interfaces and apply the same policies to them, instead of having to write the same policy for each interface. In this way, access to critical internal resources such as payroll servers or engineering code servers can be strictly controlled. Zones may be one

[3] CAC – Common Access Card

of several types: Trusted (e.g., Local Area Network (LAN)), Untrusted (e.g., WAN and virtual Multicast), Public (e.g., Demilitarized Zone (DMZ)), Encrypted (e.g., VPN), and Wireless, as well as custom zones.

- Trusted zones provide the highest level of trust. In other words, the least amount of scrutiny is applied to traffic coming from trusted zones. The LAN zone is always trusted. Conversely, traffic destined to a trusted zone is subject to the greatest scrutiny.
- Untrusted zones represent the lowest level of trust. Traffic from untrusted zones is not permitted to enter any other zone type without explicit rules, but traffic from any other zone type is permitted to enter Untrusted zones.
- Public zones offer a higher level of trust than Untrusted zones, but a lower level of trust than Trusted zones. Traffic from a Public zone to a trusted zone is denied by default. But traffic from any Trusted zone to any other zone is allowed.
- Encrypted zones are used exclusively by the VPN functionality of SonicOS Enhanced. All traffic to and from an Encrypted zone is encrypted.
- Wireless zones are zones where the only interface to the network consists of SonicWALL SonicPoint (wireless) devices. Wireless zones are not part of the evaluated configuration of the product.

SonicOS Enhanced also provides client functionality for Domain Name System (DNS) resolution, Address Resolution Protocol (ARP), and Network Address Translation (NAT). It includes a Network Time Protocol (NTP) client that automatically adjusts the product's clock, which provides time stamps for log events, automatic updates to services, and other internal purposes. The System Time will be set to no automatic update using NTP for the evaluation.

An administrator manages SonicOS Enhanced through a web management GUI, using Hypertext Transfer Protocol (HTTP) or HTTPS and a web browser. All management activities can be performed through the web management GUI, via a hierarchy of menu buttons. These activities include:

- Dashboard: The Visualization Dashboard allows administrators to monitor the network, logs, connections, and applications.
- System: Security appliance controls such as managing system status, managing licenses, configuring remote management options, managing firmware versions and preferences, and troubleshooting diagnostic tools.
- Network: Configure logical interfaces, load balancing, failover, security zones, address objects, routing, the DHCP server, IP Helper, web proxy server, and dynamic DNS. Creation of NAT policies and setting up DNS servers is also available.
- Third Generation (3G)/Analog Modem, Wireless, and SonicPoint: Different pages on wireless functionality, which is excluded from the TOE.
- Firewall and Firewall Settings: Establish access rules.
- DPI-Secure Socket Layer (SSL): Allow DPI of encrypted HTTPS traffic. This functionality is not included in the evaluation.
- Voice over IP (VoIP): Setup and configuration of Session Initiated Protocol (SIP) Voice over IP using IPsec VPNs.
- Anti-Spam: Configure the anti-spam feature.
- VPN: Create VPN policies and creating site-to-site VPN policies.
- User Management: Configure appliances for user level authentication.
- High Availability: Configure high availability settings.
- Security Services: Activate security services and use of Intrusion Protection Service, Content Filtering, and Client Anti-Virus.
- Log: Manage the logs and alerts for the system.

The product has five modes of operation: Layer 2 Bridged (L2B) Mode, Transparent Mode, IPS Sniffer Mode, Wire Mode, and Central-site Gateway Mode. Multiple modes of operation can exist simultaneously. When the appliance is deployed in Central-site Gateway Mode each interface can provide typical routing

functionality. Transparent Mode allows a SonicWALL appliance to be introduced into a network without the need for re-addressing. Layer 2 Bridged mode can be set up in the regular mode, which is similar to Transparent Mode, or in the IPS Sniffer Mode. IPS Sniffer Mode mirrors the traffic from a network switch to examine network traffic. Wire Mode has four setting: Bypass Mode, Inspect Mode, Secure Mode, and Tap Mode each of which is explained in the *SonicWALL SonicOS Enhanced v5.9 Administrator's Guide*.

1.4 TOE Overview

The TOE Overview summarizes the usage and major security features of the TOE. The TOE Overview provides a context for the TOE evaluation by identifying the TOE type, describing the product, and defining the specific evaluated configuration.

The TOE is one or more firewall/UTM/VPN that runs on a TZ 105, TZ 105W, TZ 205, TZ 205W, TZ 215, TZ 215W, NSA 220, NSA 220W, NSA 240, NSA 250M, NSA 250MW, NSA 2400, NSA 2400MX, NSA 3500, NSA 4500, NSA E5500, NSA E6500, NSA E7500, NSA E8500, NSA E8510, NSA E10400, NSA E10800, and NSA E10200 SonicWALL appliance. The appliance is installed on a network wherever firewall/UTM/VPN services are required, as depicted in Figure 1 below. This may be used at the edge of a network for perimeter security or between different segments of a network for internal security. The TOE is a hardware TOE that includes SonicOS Enhanced v5.9.0 running on each of the hardware appliances listed above.

The TOE includes all of the components and functionality described above and in section 1.5, except for the features and functionality listed below in section 1.5.3. Section 1.4.1 identifies any major non-TOE hardware and software that is required by the TOE.

Figure 1 shows the details of the deployment configuration of the TOE:

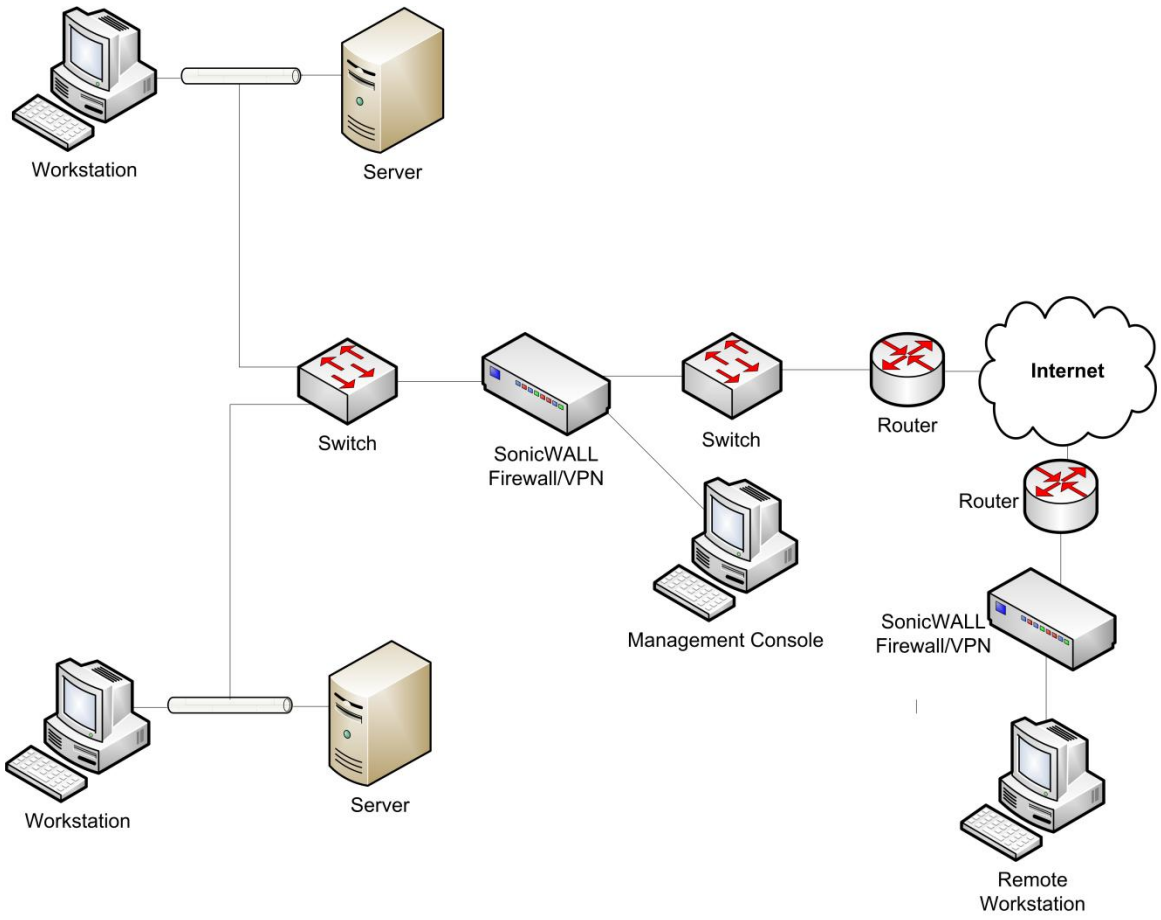

Figure 1 Deployment Configuration of the TOE

1.4.1 TOE Environment

The TOE environment consists of a management console for managing the TOE and the LAN to which the TOE is connected. The minimum system requirements for the proper operation of the TOE are:

- Management Console – General purpose computer with
  - Chrome 4.0 or higher (recommended browser)
  - Mozilla 3.0 or higher, or
  - Internet Explorer 8.0 or higher

  for HTTPs management sessions.
- LAN and associated switches and routers configured to use SonicWALL appliance as a gateway or layer 2 bridge.

In addition, the TOE needs cable and connectors that allow all of the TOE and environmental components to communicate with each other.

It is assumed that there will be no untrusted users or software on the TOE. In addition, the TOE is intended to be deployed in a physically secured cabinet, room, or data center with only authorized individuals allowed physical access to the appliance.

1.5 TOE Description

This section primarily addresses the physical and logical components of the TOE included in the evaluation. The TOE is a combined hardware and software TOE.

1.5.1 Physical Scope

Figure 2 illustrates the physical scope and the physical boundary of the overall solution and ties together all of the components of the TOE and the constituents of the TOE Environment. The TOE is a hardware TOE and consists of SonicOS Enhanced v5.9.0 and one of the following hardware appliances: TZ 105, TZ 105W, TZ 205, TZ 205W, TZ 215, TZ 215W, NSA 220, NSA 220W, NSA 240, NSA 250M, NSA 250MW, NSA 2400, NSA 2400MX, NSA 3500, NSA 4500, NSA E5500, NSA E6500, NSA E7500, NSA E8500, NSA E8510, NSA E10400, NSA E10800, and NSA E10200. The TZ appliances are designed for small businesses, with lower throughput and fewer interfaces than the NSA models. The NSA models offer multi-core architecture, with higher throughput and more interfaces. The NSA E-Class models are designed for high-speed access and heavy workgroup segmentation. They also offer multi-core architecture, high throughput and increased interfaces. All appliances run Sonic OS Enhanced and all appliances can perform the functionality of the TZ appliances. The NSA and NSA E-Class appliances do provide additional functionality, but this additional functionality is excluded from this evaluation.

The TOE is a UTM which runs on the SonicWALL NSA series and TZ series hardware appliances listed above. The TOE is installed on a network wherever firewall/UTM/VPN services are required, as depicted in the figure below. The essential physical components for the proper operation of the TOE in the evaluated configuration are

- SonicOS Enhanced v5.9.0
- SonicWALL appliance hardware

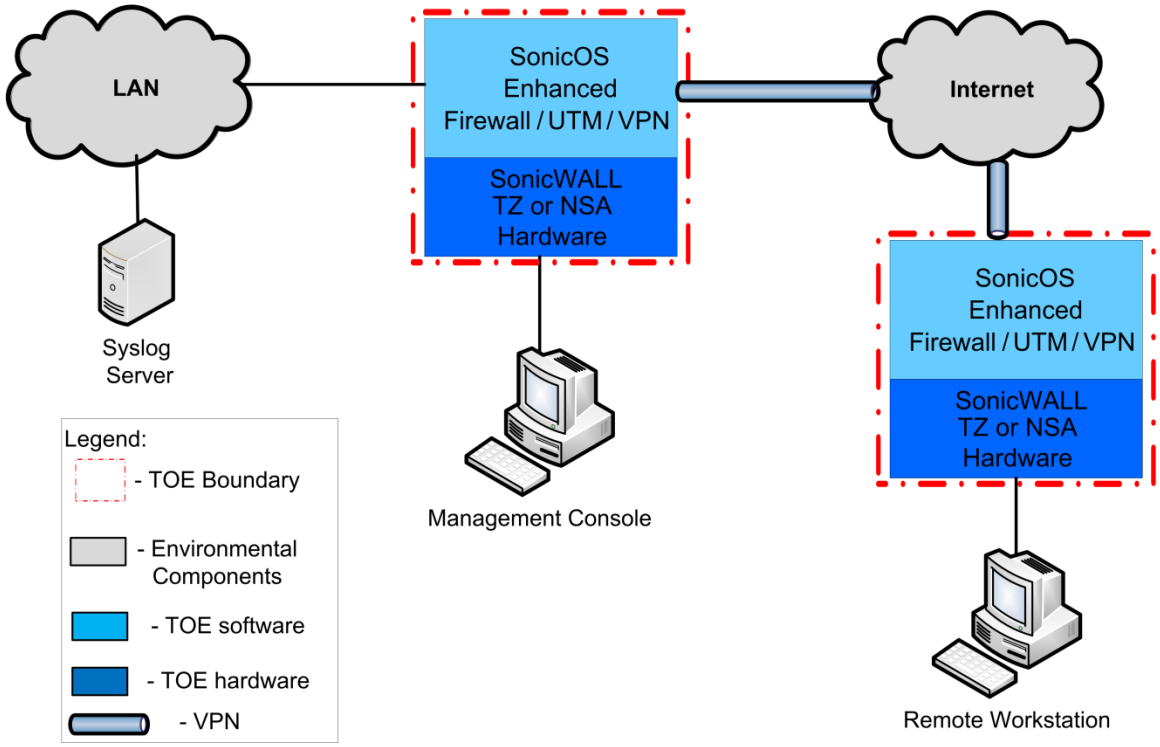

Figure 2 Physical TOE Boundary

The TOE Boundary includes all the SonicWALL developed parts of the SonicOS Enhanced v5.9.0 product. Any modified third party source code or software that SonicOS Enhanced v5.9.0 includes is considered to be TOE Software.

1.5.1.1 Guidance Documentation

Table 2 lists the TOE Guidance Documentation to install, configure, and maintain the TOE.

Table 2 Guidance Documentation

Document Name	Description
Dell SonicWALL, Inc. NSA 220 Series Quick Start Poster P/N 232-002003-50 Rev A 09/11 Dell SonicWALL, Inc. NSA 240 Getting Started Guide P/N 232-001580-00 Rev A 2/2010 Dell SonicWALL, Inc. NSA 250M or 250MW Quick Start Poster P/N 232-001924-51 Rev A 01/12 Dell SonicWALL, Inc. NSA 2400 Getting Started Guide P/N 232-001276-52 Rev A 04/10 Dell SonicWALL, Inc. NSA 2400MX Getting Started Guide P/N 232-001475-51 Rev A 3/10 Dell SonicWALL, Inc. NSA 5000/4500/3500 Getting Started Guide P/N 232-001265-52 Rev A 01/11 Dell SonicWALL, Inc. NSA E5500 Getting Started Guide P/N 232-001052-55 Rev A 3/13 Dell SonicWALL, Inc. NSA E6500 Getting Started Guide P/N 232-001051-54 Rev A 03/13 Dell SonicWALL, Inc. NSA E7500 Getting Started Guide P/N 232-001050-55 Rev A 04/13 Dell SonicWALL, Inc. NSA E8500 Getting Started Guide P/N 232-001891-53 Rev A 04/13 Dell SonicWALL, Inc. NSA E8510 Getting Started Guide P/N 232-001858-51 Rev A 04/13 Dell SonicWALL, Inc. TZ 105 Series Quick Start Poster P/N 232-002038-50 Rev A 03/12 Dell SonicWALL, Inc. TZ 205 Series Quick Start Poster P/N 232-002114-51 Rev A 04/12 Dell SonicWALL, Inc. TZ 215 Series Quick Start Poster P/N 232-002037-51 Rev A 11/11	Includes steps for the basic initialization and setup of the TOE.
Dell SonicWALL, Inc. SuperMassive Series Datasheet	Provides details of NSA E10200, E10400, and E10800 models.

Document Name	Description
Dell SonicWALL, Inc. SonicOS Enhanced 5.9 Administrator's Guide Rev A Dell SonicWALL, Inc. SonicOS 5.5.1.2 FIPS/Common Criteria Release Notes 232-001907-00 Rev A 07/10 Dell SonicWALL, Inc. SonicOS 5.9.0.0 Release Notes 232-000925-00 Rev D 08/13 Dell SonicWALL, Inc. SonicOS 5.9.0.0 SuperMassive 10000 Series Release Notes 232-002305-00 Rev A	Contains detailed steps for how to properly configure and maintain the TOE.

The NSA E10200, E10400, and E10800 are installed by SonicWALL professional services and therefore do not have an associated Getting Started Guide or Poster.

1.5.2 Logical Scope

The logical boundary of the TOE will be broken down into the following security classes which are further described in sections 5 and 6 of this ST. The logical scope also provides the description of the security features of the TOE. The security functional requirements implemented by the TOE are usefully grouped under the following Security Function Classes:

- Security Audit
- Cryptographic Support
- User Data Protection
- Identification and Authentication
- Security Management
- Protection of the TOE Security Function (TSF)
- TOE Access
- Trusted Path/Channels

1.5.2.1 Security Audit

Event logging by SonicOS Enhanced provides a mechanism for tracking potential security threats. Administrators can configure the log events to be automatically sent to an e-mail address for alerting, convenience, or archiving, or export the logs to an Excel file or syslog server.

The TOE generates audit records for startup and shutdown of the audit functions, blocked traffic, blocked websites, administrator account activity, VPN activity, firewall activity, firewall rule modification, network access, IPS/GAV/SPY activity, and login attempts. The audit events are associated with the administrator who performs them if applicable. Syslog audit records are transmitted over an IPsec VPN tunnel to an external syslog server in the IT environment for storage. The TOE will stop sending and an alarm is reported if this link is broken.

1.5.2.2 Cryptographic Support

The Cryptographic Support TSF function provides cryptographic functions to secure sessions between the management console's web browser and the TOE's web management GUI. HTTPS/TLS is used to secure these communications sessions. In addition, data encryption and decryption is provided by the TOE. The TOE provides IPsec VPN functionality for secure communications over the public internet. IKE protocol is used for exchanging authentication information and establishing the VPN tunnel. The TOE supports both version 1 and version 2 of IKE, but aggressive mode is not supported. Traffic within the tunnel is encrypted using AES. All TOE appliances are validated to FIPS 140-2, all cryptographic operations are performed in accordance with FIPS 140-2, and all keys, algorithms, and key destruction meet the FIPS 140-2 standard.

1.5.2.3 User Data Protection

The TOE clears memory buffers of all packet data when a packet exits so that none of the packet's data can inadvertently be exposed to another user.

1.5.2.4 Identification and Authentication

The Identification and Authentication TSF ensures that only legitimate administrators can gain access to the configuration settings and management settings of the TOE. Administrators must log in with a valid user name and password before the TOE will permit the administrators to manage the TOE. In the evaluated configuration, the TOE supports local authentication only. Prior to identification and authentication, end-users of the TOE are presented with an advisory notice and consent warning.

The TOE provides functionality that requires administrators to verify their claimed identity with password restrictions. Using password-based authentication, passwords are obscured from users when entered from the management console. Password must be composed of a combination of upper and lower case letters, numbers, and special characters.

1.5.2.5 Security Management

The TOE provides a set of commands for administrators to manage the security functions, configuration, and other features of the TOE components. The Security Management function specifies user roles with defined access for the management of the TOE components. The Security Administrator role specified by Network Device Protection Profile (NDPP) is filled by the Full Administrator in config mode.

1.5.2.6 Protection of the TSF

The TOE provides a reliable timestamp for its own use. A digital signature is used to verify all software updates that are applied to the TOE. Administrators must verify the signature prior to installing the updates. The TOE also prevents the reading of plaintext passwords and keys by hashing prior to storage as well as enforcing access control on stored critical security parameters (CSP), including symmetric keys, shared secrets, and private keys. To protect the integrity of its security functions, the TOE implements a suite of self-tests at startup. When two or more appliances are used the communications between those appliances are protected via an IPsec VPN.

1.5.2.7 TOE Access

The TOE automatically logs out a user from the web management GUI, a remote session, or a local session after an administrator-specified amount of idle time. In addition, the TOE provides administrators with the ability to terminate their own session. The web management GUI presents an administrator-defined advisory notice and consent warning message at log in.

1.5.2.8 Trusted Path/Channels

The TSF provides IPsec VPN tunnels for trusted communication between external IT[4] entities and the TOE. The TOE implements HTTPS for protection of communications between itself and the Management Console. HTTPS (TLS) connections are used to protect all communication between the TOE and management interfaces. HTTPS leverages cryptographic capabilities to protect data transfer from disclosure and modification. The management communication channels between the TOE and remote entity are distinct from other communication channels and provide assured identification of both endpoints.

1.5.3 Product Physical/Logical Features and Functionality not included in the TOE

Features/Functionality that are not part of the evaluated configuration of the TOE are:

- Command Line Interface (CLI) (Secure Shell (SSH))
- Application Firewall

[4] IT – Information Technology

- Web Content Filtering
- Lightweight Directory Access Protocol authentication
- Hardware Failover
- Real-time Blacklist (Simple Mail Transfer Protocol (SMTP))
- Global Security Client (including GroupVPN)
- Global Management System
- SonicPoint
- VoIP

2 Conformance Claims

This section provides the identification for any CC, Protection Profile (PP), and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for CC and PP conformance claims can be found in Section 8.1.

Table 3 CC and PP Conformance

Common Criteria (CC) Identification and Conformance	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 3, July 2009; CC Part 2 extended; CC Part 3 conformant; PP claim NDPP conformant; Parts 2 and 3 Interpretations of the CEM[5] as of 2012/05/01 were reviewed, and no interpretations apply to the claims made in this ST.
PP Identification	Protection Profile for Network Devices v1.1 Augmented with Flaw Remediation ALC_FLR.2, ADV_FSP.2, ADV_TDS.1, ATE_FUN.1, ASE_SPD.1, ASE_REQ.2, and ATE_COV.1

[5] Common Evaluation Methodology

3 Security Problem

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

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

3.1 Threats to Security

This section identifies the threats to the IT[6] assets against which protection is required by the TOE or by the security environment. The threat agents are divided into four categories:

- Attackers who are not TOE users: They have public knowledge of how the TOE operates and are assumed to possess a low skill level, limited resources to alter TOE configuration settings or parameters and no physical access to the TOE. Attackers are referred to as an unauthorized person, user, malicious user, malicious party in the below threats.
- TOE users: They have extensive knowledge of how the TOE operates and are assumed to possess a high skill level, moderate resources to alter TOE configuration settings or parameters and physical access to the TOE. (TOE users are, however, assumed not to be willfully hostile to the TOE.) An authorized administrator is considered to be a TOE user.
- TOE failure: The threat of the TOE failing in its operations or exhausting its resources which leads to a failure of TOE operations.
- External IT Entities: External IT entities that are being used by malicious attackers to adversely affect the security of the TOE.

All are assumed to have a low level of motivation. The IT assets requiring protection are the TSF[7] and user data saved on or transitioning through the TOE and the hosts on the protected network. Removal, diminution and mitigation of the threats are through the objectives identified in. The following threats are applicable to the TOE:

Table 4 Threats

Name	Description
T.ADMIN_ERROR	An administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized

[6] IT – Information Technology
[7] TSF – TOE Security Functionality

Name	Description
	access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.

3.2 Organizational Security Policies

An Organizational Security Policy (OSP) is a set of security rules, procedures, or guidelines imposed by an organization on the operational environment of the TOE. The OSPs for this ST are defined in Table 5.

Table 5 Organizational Security Policies

Name	Description
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.

3.3 Assumptions

This section describes the security aspects of the intended environment for the evaluated TOE. The operational environment must be managed in accordance with assurance requirement documentation for delivery, operation, and user guidance. The following specific conditions are required to ensure the security of the TOE and are assumed to exist in an environment where this TOE is employed.

Table 6 Assumptions

Name	Description
A.NO_GENERAL_PURPOSE	It is assumed that 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.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

4 Security Objectives

Security objectives are concise, abstract statements of the intended solution to the problem defined by the security problem definition. The set of security objectives for a TOE form a high-level solution to the security problem. This high-level solution is divided into two part-wise solutions: the security objectives for the TOE, and the security objectives for the TOE's operational environment. This section identifies the security objectives for the TOE and its supporting environment.

4.1 Security Objectives for the TOE

The specific security objectives for the TOE are as follows:

Table 7 Security Objectives for the TOE

Name	Description
O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the Administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.

4.2 Security Objectives for the Operational Environment

4.2.1 IT Security Objectives

The following IT security objectives are to be satisfied by the environment:

Table 8 IT Security Objectives

Name	Description
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

Name	Description
	necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

5 Extended Components

This section defines the extended SFRs and extended SARs met by the TOE. These requirements are presented following the conventions identified in Section 6.1.

5.1 Extended TOE Security Functional Components

This section specifies the extended SFRs for the TOE. The extended SFRs are organized by class. Table 9 identifies all extended SFRs implemented by the TOE.

Table 9 Extended TOE Security Functional Requirements

Name	Description
FAU_STG_EXT.1	External audit trail storage
FCS_CKM_EXT.4	Cryptographic key destruction
FCS_HTTPS_EXT.1	Extended: HTTPS
FCS_IPSEC_EXT.1	Extended: IPSEC
FCS_RBG_EXT.1	Extended: Cryptographic operation (random bit generation)
FCS_TLS_EXT.1	Extended: TLS
FIA_PMG_EXT.1	Password management
FIA_UAU_EXT.2	Extended: Password-based authentication mechanism
FIA_UIA_EXT.1	User identification and authentication
FPT_APW_EXT.1	Extended: Protection of administrator passwords
FPT_SKP_EXT.1	Extended: Management of TSF data (for reading symmetric keys)
FPT_TST_EXT.1	TSF self test
FPT_TUD_EXT.1	Extended: Trusted update
FTA_SSL_EXT.1	TSF-initiated session locking

5.1.1 Class FAU: Security Audit

Families in this class address the requirements for functions to implement security audit as defined in CC Part 2.

5.1.1.1 Family FAU_STG_EXT: Security audit event storage

Family Behaviour

This family defines the requirements for the TSF to be able to create and maintain a secure audit trail. Stored audit records refers to those records within the audit trail, and not the audit records that have been retrieved (to temporary storage) through selection. The requirements of this family are focused on the secure transmission of audit records to a remote logging server.

Components in this family address the requirements for protection audit data as defined in CC Part 2. This section defines the extended components for the FAU_STG_EXT family.

Component Leveling

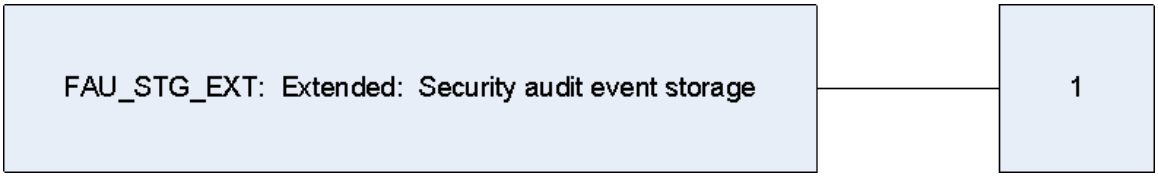

Figure 3 Extended: Security audit event storage family decomposition

The extended FAU_STG_EXT family is modeled after the FAU_STG family. FAU_STG_EXT.1 is the only component of this family.

FAU_STG_EXT.1 Extended: External audit trail storage requires the TSF to use an external IT entity for audit data storage. It was modeled after FAU_STG.1.

Management: FAU_STG_EXT.1

a) There are no management activities foreseen.

Audit: FAU_STG_EXT.1

a) There are no audit activities foreseen.

FAU_STG_EXT.1 Extended: External audit trail storage
Hierarchical to: No other components
FAU_STG_EXT.1.1
The TSF shall be able to [selection: transmit the generated audit data to an external IT entity, receive and store audit data from an external IT entity] using a trusted channel implementing the [selection: IPsec, SSH, TLS, TLS/HTTPS] protocol.
Dependencies: FTP_ITC.1 Inter-TSF trusted channel

5.1.2 Class FCS: Cryptographic Support

Families in this class address the requirements for functions to implement cryptographic functionality as defined in CC Part 2.

5.1.2.1 Family FCS_CKM_EXT: Extended: Cryptographic key management

Family Behaviour

Cryptographic keys must be managed throughout their life cycle. This family is intended to support that lifecycle and consequently defines requirements for the following activities: cryptographic key generation, cryptographic key distribution, cryptographic key access and cryptographic key destruction. This family should be included whenever there are functional requirements for the management of cryptographic keys.

Components in this family address the requirements for managing cryptographic keys as defined in CC Part 2. This section defines the extended components for the FCS_CKM_EXT family.

Component Leveling

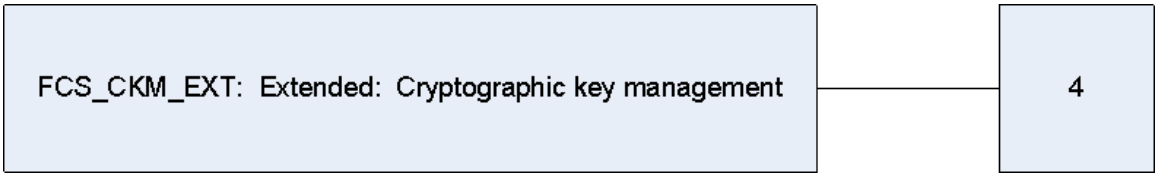

Figure 4 Extended: Cryptographic key management family decomposition

The extended FCS_CKM_EXT.4 component is the only component of the extended FCS_CKM_EXT family, which is modeled after FCS_CKM.

FCS_CKM_EXT.4 Extended: Cryptographic key zeroization, requires cryptographic keys and cryptographic critical security parameters to be zeroized. It was modeled after FCS_CKM.4

Management: FCS_CKM_EXT.4

a) There are no management activities foreseen.

Audit: FCS_CKM_EXT.4

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure on invoking the cryptographic key zeroization functionality.

FCS_CKM_EXT.4 Extended: Cryptographic key zeroization
Hierarchical to: No other components
FCS_CKM_EXT.4.1
The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.
Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

5.1.2.2 Family FCS_HTTPS_EXT: Extended: HTTPS

Family Behaviour

This family defines the requirements for protecting remote management sessions between the TOE and an authorised administrator. This family describes the how HTTPS will be implemented. The extended family "FCS_HTTPS_EXT: Extended: HTTPS" was modeled after FCS_COP: Cryptographic operation.

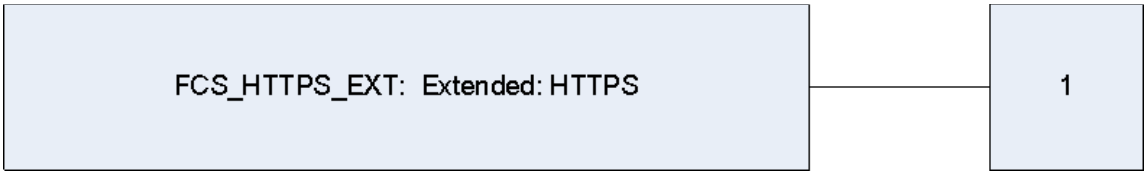

Figure 5 Extended: HTTPS family decomposition

FCS_HTTPS_EXT.1 Extended: HTTPS, requires that HTTPS be implemented according to RFC[8] 2818. This is the only component of the FCS_HTTPS_EXT family.

Management: FCS_HTTPS_EXT.1

a) There are no management activities foreseen.

Audit: FCS_ HTTPS _EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure of HTTPS session establishment

FCS_HTTPS_EXT.1 Extended: HTTPS
Hierarchical to: No other components
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 as specified in FCS_TLS_EXT.1.
Dependencies: FCS_TLS_EXT.1 Extended: TLS

[8] RFC – Request for Comment

5.1.2.3 Family FCS_IPSEC_EXT: Extended: IPSEC

Family Behaviour

Components in this family address the requirements for protecting communications using IPSEC. This is a extended family defined for the FCS Class. The extended family "FCS_IPSEC_EXT: Extended: IPSEC" was modeled after FCS_COP: Cryptographic operation.

Component Leveling

Figure 6 Extended: IPSEC family decomposition

FCS_IPSEC_EXT.1 Extended: IPSEC, requires that IPSEC be implemented as specified. FCS_IPSEC_EXT.1 is the only component of this family.

Management: FCS_IPSEC_EXT.1

a) There are no management activities foreseen.

Audit: FCS_ IPSEC _EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure of establishment of an IPSEC security association (SA).

FCS_IPSEC_EXT.1 Extended: IPSEC
Hierarchical to: No other components
FCS_IPSEC_EXT.1.1
The TSF shall implement the IPsec protocol ESP[9] as defined by RFC 4303 using the cryptographic algorithms AES[10]-CBC[11]-128, AES-CBC-256 (both specified by RFC 3602), [selection: no other algorithms, AES-GCM[12]-128, AES-GCM-256 as specified in RFC 4106], and using [selection: choose at least one of: IKEv1 as defined in RFCs 2407, 2408, 2409, and 4109, and [selection: no other RFCs for hash functions, RFC 4868 for hash functions]; IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23), 4307, and [selection: no other RFCs for hash functions, RFC 4868 for hash functions]].
FCS_IPSEC_EXT.1.2
The TSF shall ensure that IKEv1 Phase 1 exchanges use only main mode.
FCS_IPSEC_EXT.1.3
The TSF shall ensure that IKEv1 SA lifetimes are able to be limited to 24 hours for Phase 1 SAs and 8 hours for Phase 2 SAs.
FCS_IPSEC_EXT.1.4
The TSF shall ensure that IKEv1 SA lifetimes are able to be limited to [assignment: *number between 100 – 200*] MB[13] of traffic for Phase 2 SAs.
FCS_IPSEC_EXT.1.5

[9] ESP – Encapsulating Security Payload
[10] AES – Advanced Encryption Standard
[11] CBC – Cipher Block Chaining
[12] GCM – Galois/Counter Mode
[13] MB - Megabyte

The TSF shall ensure that all IKE protocols implement DH[14] Groups 14 (2048-bit MODP[15]), and [selection: 24 (2048-bit MODP with 256-bit POS[16]), 19 (256-bit Random ECP[17]), 20 (384-bit Random ECP), [assignment: *other DH groups that are implemented by the TOE*], no other DH groups].

FCS_IPSEC_EXT.1.6

The TSF shall ensure that all IKE protocols implement Peer Authentication using the [selection: DSA[18], rDSA[19], ECDSA[20]] algorithm.

FCS_IPSEC_EXT.1.7

The TSF shall support the use of pre-shared keys (as referenced in the RFCs) for use in authenticating its IPsec connections.

FCS_IPSEC_EXT.1.8

The TSF shall support the following:

1. Pre-shared keys shall be able to be composed of any combination of upper and lower case letters, numbers, and special characters: [selection: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")"];
2. Pre-shared keys of 22 characters and [selection: [assignment: *other supported lengths*], no other lengths].

Dependencies: **FCS_COP.1 Cryptographic operation**

[14] DH – Diffie-Hellman
[15] MODP –Exponentiation Modulo a Prime
[16] POS – Prime Order Subgroup
[17] ECP – Elliptic Curve modulo a Prime
[18] DSA – Digital Signature Algorithm
[19] rDSA – RSA Digital Signature Algorithm
[20] ECDSA – Elliptic Curve Digital Signature Algorithm

5.1.2.4 Family FCS_RBG_EXT: Extended: Random Bit Generation

Family Behaviour

Components in this family address the requirements for random number / bit generation. This is a new family defined for the FCS Class, and is modeled after FCS_COP: Cryptographic operation.

Component Leveling

Figure 7 Extended: Random Bit Generation family decomposition

FCS_RBG_EXT.1 Extended: Random Bit Generation, requires random bit generation to be performed in accordance with selected standards and seeded by an entropy source. It was modeled after FCS_COP.1. FCS_RBG_EXT.1 is the only component of this family.

Management: FCS_RBG_EXT.1

a) There are no management activities foreseen.

Audit: FCS_RBG_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) There are no auditable events foreseen.

FCS_RBG_EXT.1 Extended: Random Bit Generation
Hierarchical to: No other components
FCS_RBG_EXT.1.1
The TSF shall perform all random bit generation (RBG) services in accordance with [selection, choose one of: NIST[21] Special Publication 800-90 using [selection: Hash_DRBG[22] (any), HMAC[23]_DRBG (any), CTR[24]_DRBG (AES), Dual_EC[25]_DRBG (any)]; FIPS Pub 140-2 Annex C: X9.31 Appendix 2.4 using AES] seeded by an entropy source that accumulated entropy from [selection: a software-based noise source; a TSF-hardware-based noise source].
FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded with a minimum of [selection, choose one of: 128 bits, 256 bits] of entropy at least equal to the greatest bit length of the keys and authorization factors that it will generate.
Dependencies: None.

[21] NIST – National Institute of Standards and Technology
[22] DRGB – Deterministic Random Bit Generator
[23] HMAC – Hashed Message Authentication Code
[24] CTR – Counter Mode
[25] EC – Elliptical Curve

5.1.2.6 Family FCS_TLS_EXT: Extended: TLS

Family Behaviour

Components in this family address the requirements for protecting communications using TLS. This is a new family defined for the FCS Class, and is modeled after FCS_COP: Cryptographic operation.

Component Leveling

Figure 8 Extended: TLS family decomposition

FCS_TLS_EXT.1 Extended: TLS, requires that TLS be implemented. FCS_TLS_EXT.1 is the only component of this family.

Management: FCS_TLS_EXT.1

a) There are no management activities foreseen.

Audit: FCS_ TLS _EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure of establishment of a TLS session.

FCS_TLS_EXT.1 Extended: TLS
Hierarchical to: No other components
FCS_TLS_EXT.1.1

The TSF shall implement one or more of the following protocols [selection: TLS 1.0 (RFC 2246), TLS 1.1 (RFC 4346), TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA
TLS_DHE_RSA_WITH_AES_256_CBC_SHA

[selection:
None
TLS_RSA_WITH_AES_128_CBC_SHA256
TLS_RSA_WITH_AES_256_CBC_SHA256
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
].

Dependencies: FCS_COP.1 Cryptographic operation

5.1.3 Class FIA: Identification and Authentication

Families in this class address the requirements for functions to establish and verify a claimed user identity as defined in CC Part 2.

5.1.3.1 Family FIA_PMG_EXT: Extended: Password Management

Family Behaviour

This family defines the password strength rules enforced by the TSF.

This section defines the extended components for the FIA_PMG_EXT family, which is modeled after FIA_SOS Specification of secrets.

Component Leveling

Figure 9 Extended: Password Management family decomposition

The extended FIA_PMG_EXT.1 component is the only component of the FIA_PMG_EXT family.

FIA_PMG_EXT.1 defines the password strength requirements that the TSF will enforce.

Management: FIA_PMG_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FIA_PMG_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) There are no auditable events foreseen.

FIA_PMG_EXT.1 Extended: Password management
Hierarchical to: No other components.
FIA_PMG_EXT.1.1

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

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [selection: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". [assignment: *other characters*]];
2. Minimum password length shall settable by the Security Administrator, and support passwords of 15 characters or greater.

Dependencies: No dependencies

5.1.3.2 Family FIA_UAU_EXT: Extended: User authentication

Family Behaviour

This family defines the types of user authentication mechanisms supported by the TSF.

This section defines the extended components for the FIA_UAU_EXT family, which is modeled after the FIA_UAU User authentication family.

Component Leveling

Figure 10 Extended: User authentication family decomposition

The extended FIA_UAU_EXT.2 Extended: Password-based authentication mechanism component is modeled after FIA_UAU.5 as defined in CC Part 2. FIA_UAU_EXT.2 is the only component of this family.

FIA_UAU_EXT.2 requires a local password-based authentication mechanism and the capability for passwords to expire. In addition, other authentication mechanisms can be specified.

Management: FIA_UAU_EXT.2

The following actions could be considered for the management functions in FMT:

b) Reset a user password by an administrator.

Audit: FIA_UAU_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) All use of the authentication mechanisms.

FIA_UAU_EXT.2 Extended: Password-based authentication mechanism
Hierarchical to: No other components
FIA_UAU_EXT.2.1
The TSF shall provide a local password-based authentication mechanism, [selection: [assignment: *other authentication mechanism(s)*], none] to perform user authentication.
Dependencies: No dependencies.

5.1.3.3 Family FIA_UIA_EXT: Extended: User identification and authentication

Family Behaviour

This family defines the types of user identification and authentication mechanisms supported by the TSF.

This section defines the components for the extended FIA_UIA_EXT family, which is modeled after FIA_UAU User authentication, and FIA_UID User identification.

Component Leveling

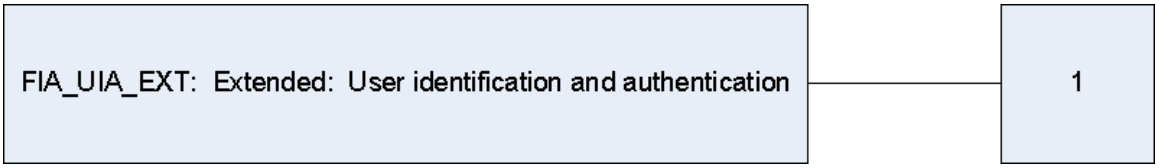

Figure 11 Extended: User identification and authentication family decomposition

The extended FIA_UIA_EXT.1 component is considered to be part of the FIA_UIA_EXT family and is based on a combination of FIA_UAU.1 and FIA_UID.1. It is the only component of this family.

FIA_UIA_EXT.1 User identification and authentication, requires a local password-based authentication mechanism and the capability for passwords to expire. In addition, other authentication mechanisms can be specified.

Management: FIA_UIA_EXT.1

The following actions could be considered for the management functions in FMT:

a) Management of the authentication data by an administrator;
b) Management of the authentication data by the associated user;
c) Managing the list of actions that can be taken before the user is identified and authenticated.
d) Management of the user identities;

Audit: FIA_UIA_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) All use of the identification and authentication mechanism.

FIA_UIA_EXT.1 Extended: User identification and authentication
Hierarchical to: No other components
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;
- [selection: no other actions, [assignment: *list of services, actions performed by the TSF in response to non-TOE requests*]].

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.

Dependencies: No dependencies

5.1.4 Class FPT: Protection of the TSF

Families in this class address the requirements for functions providing integrity and management of mechanisms that constitute the TSF and of the TSF data as defined in CC Part 2.

5.1.4.1 Family FPT_APW: Extended: Protection of administrator passwords

Family Behaviour

Components in this family address the requirements for managing and protecting TSF data, such as passwords and keys. This is a new family modeled after the FPT Class.

Component Leveling

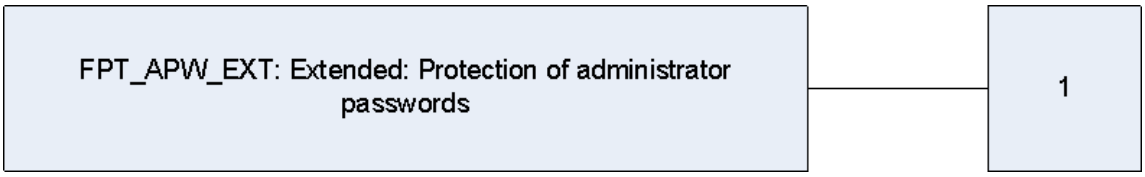

Figure 12 Extended: Protection of administrator passwords family decomposition

FPT_APW_EXT.1 Extended: Protection of administrator passwords, requires preventing selected TSF data from being read by any user or subject.

Management: FPT_APW_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FPT_APW_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) There are no auditable events foreseen.

FPT_APW_EXT.1 Extended: Protection of administrator passwords
Hierarchical to: No other components
Dependencies: No dependencies.
FPT_APW_EXT.1.1
The TSF shall store passwords in non-plaintext form.
FPT_APT_EXT.1.2
The TSF shall prevent the reading of plaintext passwords.

5.1.4.2 Family FPT_SKP: Extended: Protection of TSF data

Family Behaviour

Components in this family address the requirements for managing and protecting TSF data, such as passwords and keys. This is a new family modeled after the FPT Class.

Component Leveling

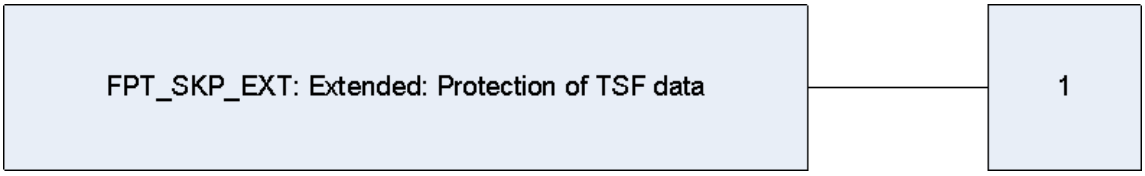

Figure 13 Extended: Protection of TSF data family decomposition

FPT_SKP_EXT.1 Extended: Protection of TSF data (for reading all symmetric keys), requires preventing symmetric keys from being read by any user or subject.

Management: FPT_SKP_EXT.1

The following actions could be considered for the management functions in FMT:

b) There are no management activities foreseen.

Audit: FPT_SKP_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

b) There are no auditable events foreseen.

FPT_SKP_EXT.1 Extended: Protection of TSF data (for reading of all symmetric keys)
Hierarchical to: No other components
Dependencies: No dependencies.
FPT_SKP_EXT.1.1
The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.1.4.3 Family FPT_TST: TSF self test

Family Behaviour

Components in this family address the requirements for self-testing the TSF for selected correct operation.

The extended FPT_TST_EXT.1 component is the only component of the FPT_TST_EXT extended family, which is modeled after FPT_TST.

Component Leveling

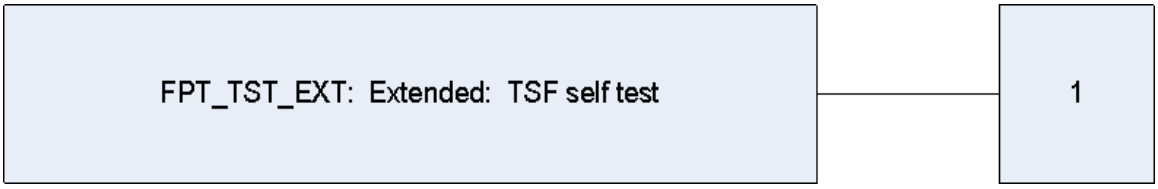

Figure 14 Extended: TSF self test family decomposition

FPT_TST_EXT.1 Extended: TSF testing, requires a suite of self tests to be run during initial start-up in order to demonstrate correct operation of the TSF. This SFR is modeled after FPT_TST.1, and is the only component of this family.

Management: FPT_TST_EXT.1

a) There are no management activities foreseen.

Audit: FPT_TST_EXT.1

a) There are no auditable activities foreseen.

FPT_TST_EXT.1 Extended: TSF testing
Hierarchical to: No other components
FPT_TST_EXT.1.1
The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.
Dependencies: No dependencies.

5.1.4.4 Family FPT_TUD: Extended: Trusted Update

Components in this family address the requirements for updating the TOE firmware and/or software. This is a new family defined for the FPT Class.

Component Leveling

Figure 15 Extended: Trusted Update family decomposition

FPT_TUD_EXT.1 Extended: Trusted update, requires management tools be provided to update the TOE firmware and software, including the ability to verify the updates prior to installation.

Management: FPT_ TUD_EXT.1

a) There are no management activities foreseen.

Audit: FPT_ TUD_EXT.1

a) Initiation of update.

FPT_ TUD_EXT.1 Extended: Trusted update
Hierarchical to: No other components
FPT_TUD_EXT.1.1
The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.
FPT_TUD_EXT.1.2
The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.
FPT_TUD_EXT.1.3
The TSF shall provide a means to verify firmware/software updates to the TOE using a [selection: digital signature mechanism, published hash] prior to installing those updates.
Dependencies: FCS_COP.1 Cryptographic operation.

5.1.5 Class FTA: TOE Access

Family Behaviour

Families in this class address the requirements for functions that control the establishment and existence of a user session as defined in CC Part 2.

5.1.5.1 Family FTA_SSL_EXT: Extended: TSF-initiated session locking

Components in this family address the requirements for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.

The extended FTA_SSL_EXT.1 component is the only component of the FTA_SSL_EXT family.

Component Leveling

Figure 16 Extended: TSF-initiated Session locking family decomposition

FTA_SSL_EXT.1 Extended: TSF-initiated session locking, requires system initiated locking of an interactive session after a specified period of inactivity.

Management: FTA_SSL_EXT.1

The following actions could be considered for the management functions in FMT:

a) Specification of the time of user inactivity after which lock-out occurs for an individual user.

Audit: FTA_SSL_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Any attempts at unlocking an interactive session.

FTA_SSL_EXT.1 Extended: TSF-initiated session locking
Hierarchical to: No other components
FTA_SSL_EXT.1.1
The TSF shall, for local interactive sessions, [selection:

- lock the session – disable any activity of the user's data access display devices other than unlocking the session, and requiring that the administrator re-authenticate to the TSF prior to unlocking the session;
- terminate the session]

after a Security Administrator-specified time period of inactivity.
Dependencies: FIA_UAU.1 Timing of authentication.

5.2 Extended TOE Security Assurance Components

There are no extended TOE Security Assurance Components.

6 Security Requirements

This section defines the SFRs and SARs met by the TOE. These requirements are presented following the conventions identified in Section 6.1.

6.1 Conventions

There are several font variations used within this ST. Selected presentation choices are discussed here to aid the Security Target reader.

The CC allows for assignment, refinement, selection and iteration operations to be performed on security functional requirements. All of these operations are used within this ST. These operations are performed as described in Part 2 of the CC, and are shown as follows:

- Completed assignment statements are identified using [*italicized text within brackets*].
- Completed selection statements are identified using [underlined text within brackets]. In keeping with these conventions, in the event an assignment is within a selection, it will be depicted as *italicized, underlined* text.
- Refinements are identified using **bold text**. Any text removed is stricken (Example: ~~TSF Data~~) and should be considered as a refinement. In keeping with these conventions, in the event a refinement is within an assignment, it will be depicted as ***bold italicized*** text, and when a refinement is within a selection, it will be depicted in **bold underlined** text.
- Extended Functional and Assurance Requirements are identified using "_EXT" at the end of the short name.
- Iterations are identified by appending a number in parentheses following the component title. For example, FAU_GEN.1(1) Audit Data Generation would be the first iteration and FAU_GEN.1(2) Audit Data Generation would be the second iteration.

6.2 Security Functional Requirements

This section specifies the SFRs for the TOE. This section organizes the SFRs by CC class. Table 10 identifies all SFRs implemented by the TOE and indicates the ST operations performed on each requirement.

Table 10 TOE Security Functional Requirements

Name	*Description*	*S*	*A*	*R*	*I*
FAU_GEN.1	*Audit data generation*	✓	✓	✓	
FAU_GEN.2	*User identity association*				
FAU_STG_EXT.1	*Extended: External audit trail storage*	✓			
FCS_CKM.1	*Cryptographic key generation*	✓		✓	
FCS_CKM_EXT.4	*Extended: Cryptographic key destruction*				
FCS_COP.1(1)	*Cryptographic operation (for data encryption/decryption)*	✓	✓	✓	✓
FCS_COP.1(2)	*Cryptographic operation (for cryptographic signature)*	✓		✓	✓
FCS_COP.1(3)	*Cryptographic operation (for cryptographic hashing)*	✓	✓	✓	✓
FCS_COP.1(4)	*Cryptographic operation (for keyed-hash message authentication)*	✓	✓	✓	✓
FCS_HTTPS_EXT.1	*Extended: HTTPS*				

Name	Description	S	A	R	I
FCS_IPSEC_EXT.1	Extended: IPSEC	✓	✓	✓	
FCS_RBG_EXT.1	Extended: Cryptographic operation (random bit generation)	✓			
FCS_TLS_EXT.1	Extended: TLS	✓			
FDP_RIP.2	Full residual information protection	✓			
FIA_PMG_EXT.1	Extended: Password management				
FIA_UIA_EXT.1	Extended: User identification and authentication	✓	✓		
FIA_UAU_EXT.2	Extended: Password-based authentication mechanism	✓	✓		
FIA_UAU.7	Protected authentication feedback		✓		
FMT_MTD.1	Management of TSF data	✓	✓		
FMT_SMF.1	Specification of management functions	✓	✓		
FMT_SMR.2	Restrictions on security roles	✓	✓		
FPT_APW_EXT.1	Extended: Protection of administrator passwords				
FPT_ITT.1	Basic Internal TSF Data Transfer Protection	✓	✓	✓	
FPT_SKP_EXT.1	Extended: Management of TSF data (for reading of all symmetric keys)				
FPT_STM.1	Reliable time stamps			✓	
FPT_TST_EXT.1	Extended: TSF testing				
FPT_TUD_EXT.1	Extended: Trusted update	✓			
FTA_SSL_EXT.1	Extended: TSF-initiated session locking	✓			
FTA_SSL.3	TSF-initiated termination		✓	✓	
FTA_SSL.4	User-initiated termination				
FTA_TAB.1	Default TOE access banners			✓	
FTP_ITC.1	Inter-TSF trusted channel		✓	✓	
FTP_TRP.1	Trusted path	✓	✓	✓	

Note: S=Selection; A=Assignment; R=Refinement; I=Iteration

6.2.1 Class FAU: Security Audit

FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

FAU_GEN.1.1

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

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

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

c) [*All administrative actions*]

d) [*Specifically defined auditable events listed in Table 11*].

Table 11 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_EXT.4	None.	None.
FCS_COP.1(1)	None.	None.
FCS_COP.1(2)	None.	None.
FCS_COP.1(3)	None.	None.
FCS_COP.1(4)	None.	None.
FCS_RBG_EXT.1	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session. Establishment/Termination of a HTTPS session.	Reason for failure. Authentication failures are the only auditable failures. Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_TLS_EXT.1	Failure to establish a TLS Session. Establishment/Termination of a TLS session.	Reason for failure. Authentication failures are the only auditable failures. Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA. Establishment/Termination of an IPsec SA.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FDP_RIP.2	None.	None.
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FMT_MTD.1	None.	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_ITT.1	None.	None.
FPT_STM.1	Changes to the time.	The old and new values for the time Origin of the attempt (e.g. IP address)
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FPT_TST_EXT.1	None.	None.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	No additional information.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_SSL.4	The termination of an interactive session.	No additional information.
FTA_TAB.1	None.	None.
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted path functions.	Identification of the claimed user identity.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempts.

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 (if applicable), 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, [*information specified in column three of Table 11*].

Dependencies: FPT_STM.1 Reliable time stamps

FAU_GEN.2 User identity association

Hierarchical to: No other components.

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.

Dependencies: FAU_GEN.1 Audit data generation
FIA_UID.1 Timing of identification

FAU_STG_EXT.1 Extended: External audit trail storage

Hierarchical to: No other components.

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 implementing the [IPsec] protocol.

Dependencies: FAU_GEN.1 Audit data generation
FTP_ITC.1 Inter-TSF trusted channel

6.2.2 Class FCS: Cryptographic Support

FCS_CKM.1 Cryptographic key generation
Hierarchical to: No other components.
FCS_CKM.1.1
The TSF shall generate **asymmetric** cryptographic keys **used for key establishment** in accordance with ~~a specified cryptographic key generation algorithm~~: [
- ***NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography" for RSA-based key establishment schemes.***

] and specified cryptographic key sizes [*equivalent to, or greater than, a symmetric key strength of 112 bits*] ~~that meet the following: [*assignment: list of standards*]~~.
Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction

FCS_CKM_EXT.4 Extended: Cryptographic key destruction
Hierarchical to: No other components.
FCS_CKM_EXT.4.1
The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.
Dependencies: FCS_CKM.1 Cryptographic key generation

FCS_COP.1(1) Cryptographic operation (for data encryption/decryption)
Hierarchical to: No other components.
FCS_COP.1.1(1)
The TSF shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm [*AES operating in* **[*CBC mode*]**] and cryptographic key sizes [*128-bits, 256-bits, and* **[*192-bits*]**] that meets the following:
- **[*FIPS PUB 197, "Advanced Encryption Standard" (AES)***
- **[*NIST SP 800-38A*]]**

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1(2) Cryptographic operation (for cryptographic signature)
Hierarchical to: No other components.
FCS_COP.1.1(2)
The TSF shall perform **cryptographic signature services** in accordance with a ~~specified cryptographic algorithm~~ [***RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater***] ~~and cryptographic key sizes [*assignment: cryptographic key sizes*]~~ that meets the following: [***FIPS PUB 186-3, "Digital Signature Standard", FIPS PUB 186-2, "Digital Signature Standard"***].
Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
Application Note: FCS_COP.1(2) has been modified from the NDPP text to include the portions from the CEM that were stricken as part of the refinement.

FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)
Hierarchical to: No other components.
FCS_COP.1.1(3)
The TSF shall perform [*cryptographic hashing services*] in accordance with a specified cryptographic algorithm [***SHA[26]-1, SHA-256***] and ~~cryptographic key sizes [*assignment:*~~

[26] SHA – Secure Hash Algorithm

~~*cryptographic key sizes*~~] **message digest sizes** [**160, 256**] **bits** that meet the following: [*FIPS Pub 180-3, "Secure Hash Standard"*].

Dependencies: **FCS_CKM.1 Cryptographic key generation**
FCS_CKM.4 Cryptographic key destruction

Application Note: FCS_COP.1(3) has been modified from the NDPP text to include the portions from the CEM that were stricken as part of the refinement.

FCS_COP.1(4) Cryptographic operation (for keyed-hash message authentication)

Hierarchical to: No other components.

FCS_COP.1.1(4)

The TSF shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm [*HMAC-* **[*SHA-256*]**], ~~and cryptographic key sizes~~ **key sizes [*20-bits, 4-bits*], and message digest sizes** [**256**] **bits** that meet the following: [*FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code, and FIPS Pub 180-3, "Secure Hash Standard"*].

Dependencies: **FCS_CKM.1 Cryptographic key generation**
FCS_CKM.4 Cryptographic key destruction

Application Note: FCS_COP.1(4) has been modified from the NDPP text to include the portions from the CEM that were stricken as part of the refinement.

FCS_HTTPS_EXT.1 Extended: HTTPS

Hierarchical to: No other components.

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 HTTPS using TLS as specified in FCS_TLS_EXT.1.

Dependencies: FCS_TLS_EXT.1 Extended: TLS

FCS_IPSEC_EXT.1 Extended: IPSEC

Hierarchical to: No other components.

FCS_IPSEC_EXT.1.1

The TSF shall implement IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms AES-CBC-128, AES-CBC-256 (both specified by RFC 3602), [no other algorithms], and using [IKEv1 as defined in RFCs 2407, 2408, 2409, and RFC 4109, and [no other RFCs for hash functions]; IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23), 4307, and [no other RFCs for hash functions].

FCS_IPSEC_EXT.1.2

The TSF shall ensure that IKEv1 Phase 1 exchanges use only main mode.

FCS_IPSEC_EXT.1.3

The TSF shall ensure that IKEv1 SA lifetimes are able to be limited to 24 hours for Phase 1 SAs and 8 hours for Phase 2 SAs.

FCS_IPSEC_EXT.1.4

The TSF shall ensure that IKEv1 SA lifetimes are able to be limited to [*200*] MB of traffic for Phase 2 SAs.

FCS_IPSEC_EXT.1.5

The TSF shall ensure that all IKE protocols implement DH Groups 14 (2048-bit MODP), and [[*DH groups 2 and 5*]].

FCS_IPSEC_EXT.1.6

The TSF shall ensure that all IKE protocols implement Peer Authentication using the [rDSA] algorithm.

FCS_IPSEC_EXT.1.7

The TSF shall support the use of pre-shared keys (as referenced in the RFCs) for use in authenticating its IPsec connections.

FCS_IPSEC_EXT.1.8

The TSF shall support the following:

1. Pre-shared keys shall be able to be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "", "(", and ")");*

2. Pre-shared keys of 22 characters and [lengths up to128].

Dependencies: FCS_COP.1(2) Cryptographic operation (for cryptographic signature)
FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)
FCS_COP.1(4) Cryptographic operation (for keyed-hash message authentication)

Application Note: NDPP FCS_IPSEC_EXT.1.1 application note states that HMAC-SHA-1 is required as the hash algorithm used by IKE implementation for AES-CBC mode. The TOE implements HMAC-SHA-256 for this purpose which is a more secure algorithm.

FCS_RBG_EXT.1 Extended: Cryptographic operation (random bit generation)

Hierarchical to: No other components.

FCS_RBG_EXT.1.1

The TSF shall perform all random bit generation (RBG) services in accordance with [NIST Special Publication 800-90 using [Hash DRBG (any)]] seeded by an entropy source that accumulated entropy from at least one independent TSF-hardware-based noise source.

FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded with a minimum of [256 bits] of entropy at least equal to the greatest bit length of the keys and authorization factors that it will generate.

Dependencies: None

FCS_TLS_EXT.1 Extended: TLS

Hierarchical to: No other components.

FCS_TLS_EXT.1.1

The TSF shall implement one or more of the following protocols [TLS 1.0 (RFC 2246)] supporting the following ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA
TLS_DHE_RSA_WITH_AES_256_CBC_SHA
[*None*].

Dependencies: FCS_COP.1(1) Cryptographic operation (for data encryption/decryption)

6.2.3 Class FDP: User Data Protection

FDP_RIP.2 Full residual information protection

Hierarchical to: No other components.

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.

Dependencies: No dependencies

6.2.4 Class FIA: Identification and Authentication

FIA_PMG_EXT.1 Extended: Password management

Hierarchical to: No other components.

FIA_PMG_EXT.1.1

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

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”];
2. Minimum password length shall settable by the Security Administrator, and support passwords of 15 characters or greater.

Dependencies: No dependencies

FIA_UIA_EXT.1 Extended: User identification and authentication

Hierarchical to: No other components.

FIA_UIA_EXT.1.1

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

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

FIA_UIA_EXT.1.2

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

Dependencies: No dependencies

FIA_UAU_EXT.2 Extended: Password-based authentication mechanism

Hierarchical to: No other components.

FIA_UAU_EXT.2.1

The TSF shall provide a local password-based authentication mechanism, [none] to perform user authentication.

Dependencies: No dependencies

FIA_UAU.7 Protected authentication feedback

Hierarchical to: No other components.

FIA_UAU.7.1

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

Dependencies: FIA_UAU.1 Timing of authentication

6.2.5 Class FMT: Security Management

FMT_MTD.1 Management of TSF Data (for general TSF data)

Hierarchical to: No other components.

FMT_MTD.1.1

The TSF shall restrict the ability to [***manage as detailed in Table 12 below***] the [*TSF data*] to the [*Security Administrators* ***as detailed in Table 12 below***].

Dependencies: FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

Application Note: "Security Administrators" consists of the following roles implemented by the TOE: Full Administrator in config mode, Full Administrator in non-config mode, and Limited Administrator.

Table 12 Management of TSF Data

Operation	TSF Data	Authorized Role
Manage	Cryptographic certificates	Full Administrator in config mode
Associate with an interface	Cryptographic protocols	Full Administrator in config mode
Configure	Network	Full Administrator in config mode and Limited Administrator
Create, delete, and modify	User accounts	Full Administrators in config or non-config mode
Create, modify, and delete	Firewall rules	Full Administrators in config or non-config mode

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

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 update the TOE, and to verify the updates using* [*digital signature*] *capability prior to installing those updates;*
- *[*
  - *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 configure the cryptographic functionality.*]]

Dependencies: FCS_COP.1(2) Cryptographic operation (for cryptographic signature)

FMT_SMR.2 Security roles

Hierarchical to: No other components.

FMT_SMR.2.1

The TSF shall maintain the roles:

- [*Authorized 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

[

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

]

are satisfied.

Dependencies: FIA_UID.1 Timing of identification

Application Note: The TOE does not maintain a role of Authorized Administrator. Instead, the TOE implements the following roles: Full Administrator in config mode, Full Administrator in non-config mode, Read-only Administrator, and Limited Administrator. The role Full Administrator in config mode is granted a superset of all other role privileges, and thus, fulfills the Authorized Administrator role defined in NDPP.

6.2.6 Class FPT: Protection of the TSF

FPT_APW_EXT.1 Extended: Protection of administrator passwords
Hierarchical to: No other components.
FPT_APW_EXT.1.1
The TSF shall store passwords in non-plaintext form.
FPT_APW_EXT.1.2
The TSF shall prevent the reading of plaintext passwords.
Dependencies: No dependencies

FPT_ITT.1 Basic Internal TSF Data Transfer Protection
Hierarchical to: No other components.
FPT_ITT.1.1
The TSF shall protect TSF data from [*disclosure* ***and detect its modification***] when it is transmitted between separate parts of the TOE **through the use [IPsec]**.
Dependencies: FCS_IPSEC_EXT.1

FPT_SKP_EXT.1 Extended: Protection of TSF data (for reading of all symmetric keys)
Hierarchical to: No other components.
FPT_SKP_EXT.1.1
The TSF shall prevent reading of all pre-shared keys, symmetric key, and private keys.
Dependencies: No dependencies

FPT_STM.1 Reliable time stamps
Hierarchical to: No other components.
FPT_STM.1.1
The TSF shall be able to provide reliable time stamps **for its own use**.
Dependencies: No dependencies

FPT_TUD_EXT.1 Extended: Trusted update
Hierarchical to: No other components.
FPT_TUD_EXT.1.1
The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.
FPT_TUD_EXT.1.2
The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.
FPT_TUD_EXT.1.3
The TSF shall provide a means to verify firmware/software updates to the TOE using a [digital signature mechanism] prior to installing those updates.
Dependencies: FCS_COP.1(2) Cryptographic operation (for cryptographic signature)
FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)

FPT_TST_EXT.1 Extended: TSF testing
Hierarchical to: No other components
FPT_TST_EXT.1.1
The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.
Dependencies: No dependencies.

6.2.7 Class FTA: TOE Access

FTA_SSL_EXT.1 Extended: TSF-initiated session locking
Hierarchical to: No other components.
FTA_SSL_EXT.1.1
The TSF shall, for local interactive sessions, [terminate the session] after a Security Administrator-specified time period of inactivity.
Dependencies: FIA_UAU.1 Timing of authentication

FTA_SSL.3 TSF-initiated termination
Hierarchical to: No other components.
FTA_SSL.3.1
The TSF shall terminate **a remote** interactive session after a [*Security Administrator-configurable time interval of inactivity at the Management Console ranging from 5 to 60 minutes*].
Dependencies: No dependencies

FTA_SSL.4 User-initiated termination
Hierarchical to: No other components.
FTA_SSL.4.1
The TSF shall allow **Administrator**-initiated termination of the **Administrator's** own interactive session.
Dependencies: No dependencies

FTA_TAB.1 Default TOE access banners
Hierarchical to: No other components.
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 ~~unauthorised~~ use of the TOE.
Dependencies: No dependencies

6.2.8 Class FTP: Trusted Path/Channels

FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components.

FTP_ITC.1.1

The TSF shall **use [IPsec] to** provide a **trusted** communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [*no other capabilities*]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **disclosure and detection of modification of the channel data**.

FTP_ITC.1.2

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

FTP_ITC.1.3

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

Dependencies: No dependencies

FTP_TRP.1 Trusted Path

Hierarchical to: No other components.

FTP_TRP.1.1

The TSF shall **use [TLS/HTTPS] to** provide **a trusted** communication path between itself and [***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 detection of modification of the communicated data*].

FTP_TRP.1.2

The TSF shall permit [***remote administrators***] to initiate communication via the trusted path.

FTP_TRP.1.3(1)

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

Dependencies: No dependencies

6.3 Security Assurance Requirements

This section defines the assurance requirements for the TOE. Assurance requirements are taken from the NDPP v1.1 augmented with ADV_FSP.2, ADV_TDS.1, ALC_FLR.2, ASE_SPD.1, ATE_COV.1, and ATE_FUN.1. Table 13 below summarizes the requirements.

Table 13 Claimed Security Assurance Requirements

Assurance Requirements	
Class ASE: Security Target evaluation	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
Class ALC : Life Cycle Support	ALC_CMC.1 Labeling of the TOE
	ALC_CMS.1 TOE CM Coverage
	ALC_FLR.2 Flaw reporting procedures
Class ADV: Development	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
Class AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
Class ATE: Tests	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.1 Independent testing – conformance
Class AVA: Vulnerability assessment	AVA_VAN.1 Vulnerability survey

7 TOE Summary Specification

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

7.1 TOE Security Functions

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

Table 14 Mapping of TOE Security Functions to Security Functional Requirements

TOE Security Function	SFR ID	Description
Security Audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_STG_EXT.1	Extended: External audit trail storage
Cryptographic Support	FCS_CKM.1	Cryptographic key generation
	FCS_CKM_EXT.4	Extended: Cryptographic key destruction
	FCS_COP.1(1)	Cryptographic operation (for data encryption/decryption)
	FCS_COP.1(2)	Cryptographic operation (for cryptographic signature)
	FCS_COP.1(3)	Cryptographic operation (for cryptographic hashing)
	FCS_COP.1(4)	Cryptographic operation (for keyed-hash message authentication)
	FCS_HTTPS_EXT.1	Extended: HTTPS
	FCS_IPSEC_EXT.1	Extended: IPSEC
	FCS_RBG_EXT.1	Extended: Cryptographic operation (random bit generation)
	FCS_TLS_EXT.1	Extended: TLS
User Data Protection	FDP_RIP.2	Full residual information protection
Identification and Authentication	FIA_PMG_EXT.1	Extended: Password management
	FIA_UIA_EXT.1	Extended: User identification and authentication
	FIA_UAU_EXT.2	Extended: Password-based authentication mechanism

TOE Security Function	SFR ID	Description
	FIA_UAU.7	Protected authentication feedback
Security Management	FMT_MTD.1	Management of TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.2	Restrictions on security roles
Protection of the TSF	FPT_APW_EXT.1	Extended: Protection of administrator passwords
	FPT_ITT.1	Basic Internal TSF Data Transfer Protection
	FPT_SKP_EXT.1	Extended: Management of TSF data (for reading of all symmetric keys)
	FPT_STM.1	Reliable time stamps
	FPT_TST_EXT.1	Extended: TSF testing
	FPT_TUD_EXT.1	Extended: Trusted update
TOE Access	FTA_SSL_EXT.1	Extended: TSF-initiated session locking
	FTA_SSL.3	TSF-initiated termination
	FTA_SSL.4	User-initiated termination
	FTA_TAB.1	Default TOE access banners
Trusted path/channels	FTP_ITC.1	Inter-TSF trusted channel
	FTP_TRP.1	Trusted path

7.1.1 Security Audit

The Security Audit function provides the TOE with generation and storage of audit records. As administrators manage and configure the TOE, their activities are tracked and recorded as audit records and stored in a 32 kilobytes rolling log buffer before being exported to an external syslog server. When the buffer becomes full the logs are exported to the syslog server and flushed from the log buffer. The resulting audit records can be examined to determine which security relevant activities took place and who (i.e., which user) is responsible for those activities. The connection between the TOE and syslog server is protected via an IPsec tunnel. The TOE will stop sending syslog audit data and report an alarm if either the physical connection or the tunnel is interrupted. The records contained within the log buffer can only be accessed by authorized administrators. If an administrator's role has permissions to view the logs then the **Log** menu button will be visible. If the administrator's role does not have permission to view the logs the menu button is not visible on the web management GUI.

The TOE provides auditing of administrator actions that occur within the web management GUI. For audit events that result from actions of identified users, the TOE associates the action with the user who took the action in the logs. The TOE generates audit records for the events listed in Table 11. Included in Table 11 are auditable events from NDPP for each claimed SFR. HTTPS or TLS sessions that fail due to authentication failures are audited. No other protocol failures are audited for HTTPS and TLS sessions.
TOE Security Functional Requirements Satisfied: FAU_GEN.1, FAU_GEN.2, FAU_STG_EXT.1.

7.1.2 Cryptographic Support

The TOE uses HTTPS/TLS connections to secure management sessions between the administrator workstation and the TOE. TLS version 1.0 is used in the TOE with the following cipher suites available:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA

Chapter 59 of the *SonicWALL SonicOS v5.9 Administrator's Guide* provides details on the HTTPS/TLS handshake process and how the TOE uses certificate-based identification.

The TOE also provides IPsec VPN functionality for secure communications over the public internet. The IKE protocol as described in FCS_IPSEC_EXT.1 is used to establish the VPN tunnel. IKE protocol is used for exchanging authentication information and establishing the VPN tunnel. The TOE supports both version 1 and version 2 of IKE, but aggressive mode in IKEv1 is not supported. The traffic within the VPN tunnel is encrypted using AES in CBC mode. Part 14 of the *SonicWALL SonicOS v5.9 Administrator's Guide* provides details on configuring VPN settings. On the **VPN > Settings** page **Proposals** tab there is a box to set the Life Time for both phase 1 and phase 2. The default value of 28800 seconds is entered and can be changed by an authorized administrator. The DH Group is also configurable through this tab. Groups 2, 5, and 14 are allowed in the evaluated configuration. Page 1031 of the *SonicWALL SonicOS v5.9 Administrator's Guide* provides details on how the VPN is negotiated including the DH negotiations. A pre-shared key is entered while configuring the VPN. The secrets are shared through the use of VPN policies. When configuring a VPM policy, the **General** tab allows an administrator to select **IKE using Preshared Secret** for the authentication method. The shared secrets is manually input. HMAC-SHA-256 is used by default as the hash algorithm by the IKE implementations for AES-CBC mode. The TOE, by default establishes ESP IPsec VPN sessions with SHA-1 authentication for phase 2. By policy, users are restricted from setting this value to None and therefore prohibited from performing confidentiality-only ESP IPsec.

All SonicWALL appliances included in the TOE are validated to FIPS 140-2 (certificate # TBD). All cryptographic operations, including key generation, algorithm implementations, and key destruction methods, are performed in accordance with the FIPS 140-2 standard. The TOE implements a NIST SP 800-90 DRGB with SHA-256 for key generation (CAVP certificate #189). The TOE implements a NIST SP 800-56B section 8.2 conformant RSA-based key establishment scheme for asymmetric key establishment. SHA-1 and SHA-256 are used for secure hashing and rDSA is used for digital signatures. Table 15 details the cryptographic operations and associated algorithms used within the TOE.

Table 15 Cryptographic Operations

Cryptographic Operations	Cryptographic Algorithm	Key Sizes (bits)	Message Digest Size	Standards (Certificate #)
Symmetric encryption and decryption	AES operating in CBC mode	128,192, 256	N/A	CAVP[27] (cert # 2015) FIPS PUB 197, "Advanced Encryption Standard (AES)", NIST SP 800-38A

[27] CAVP – Cryptographic Algorithm Validation Program

Cryptographic Operations	Cryptographic Algorithm	Key Sizes (bits)	Message Digest Size	Standards (Certificate #)
Keyed-Hash Message Authentication	HMAC-SHA-256	key1 – 20, key2 – 4	256	CAVP (cert #1219) FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code", and FIPS Pub 180-3, "Secure Hash Standard"
Cryptographic hashing services	SHA-1, SHA-256		160, 256	CAVP (cert #1765) FIPS Pub 180-3, "Secure Hash Standard."
Cryptographic signature services	rDSA	2048 bit	N/A	CAVP (cert #640) Case: RSA Digital Signature Algorithm FIPS PUB 186-3, "Digital Signature Standard"

The ephemeral keys used for TLS and IPsec sessions are zerorized upon termination of the session. The IPsec shared secret is stored with the VPN policy and is zeroized if the policy is removed. Setting the TOE to factory default also zeroizes all keys. The zeroization process overwrites the memory with zeros.

The TOE relies on both hardware and software based noise sources to provide entropy for cryptographic purposes. Software-based sources are concatenated through a SHA-256 hash. The entropy provided is tested to ensure randomness using a continuous random number generator test. If an error occurs within the continuous random number generator test, the TOE enters an error state and any further entropy output is inhibited.
TOE Security Functional Requirements Satisfied: FCS_CKM.1, FCS_CKM_EXT.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_HTTPS_EXT.1, FCS_IPSEC_EXT.1, FCS_RBG_EXT.1, FCS_TLS_EXT.1.

7.1.3 User Data Protection

Residual information clearing ensures that data is not accidentally "leaked" into network packets by ensuring that packet memory buffers are cleared of packet data before reallocation. The TOE ensures that no residual data remains prior to reallocation of memory, ensuring that any attempt to reconstruct the content of the memory buffers after reallocation will result in the reconstruction of the zeros, not packet data. Once packets have left the TOE the memory buffers are freed to the buffer pool. When memory is requested from the buffer pool the memory is overwritten with zeros and then reallocated.
TOE Security Functional Requirements Satisfied: FDP_RIP.2.

7.1.4 Identification and Authentication

The web management GUI on the TOE is utilized in accessing this function. Prior to identifying and authenticating to the TOE, all users are presented with a login screen that displays and advisory notice and consent warning. The TOE must perform successful identification and authentication of the TOE user before granting access to other TOE security functions. Administrator authentication is enforced through the use of a password that complies with the rules listed in FIA_PMG_EXT.1.

In the evaluated configuration of the TOE, local authentication is the only supported mechanism for password-based authentication. Passwords are obscured with dots to prevent "shoulder surfing" by an unauthorized individual and ensure unauthorized persons cannot read the password and gain access to the TOE.
TOE Security Functional Requirements Satisfied: FIA_PMG_EXT.1, FIA_UIA_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7.

7.1.5 Security Management

Security management specifies how the TOE manages several aspects of the TSF including TSF data and security functions. TSF data includes configuration data of the TSF, audit data, and cryptographic information and protocols. For a listing of TSF data managed by the TOE see Table 12. Management of security functions includes configuring syslog forwarding, cryptographic protocols, firewall rules, authentication, password complexity requirements, login banners, session timeout, and system time, as well as performing system updates. The TOE provides authorized administrators with a web management GUI to manage the security functions and TSF data of the TOE. The TOE displays a warning banner to users and administrators prior to log in. No other functions can be performed prior to log in.

Once a user has logged into the TOE their session is bound to the permissions associated with their username. The TOE verifies permissions when a command is sent and only executes the command if the user has the role associated with each operation in Table 12. Only a Full Administrator in config mode can perform cryptographic operations such as, managing certificates and associating a cryptographic protocol with an interface. Full Administrators in both config mode and non-config mode can manage user accounts. Only a Full Administrator in config mode and a Limited Administrator can configure the network.

The TOE supports the following administrator roles:

- Read-only Administrator – Have read-only access to view the management interface and download troubleshooting reports.
- Limited Administrator – Have the access of a Read-only Administrator and can manage logs, configure the network, and perform diagnostics.
- Full Administrator – Members of this group have all the privileges of a Limited Administrator and can manage users, configure the TOE, and manage cryptography. Only one Full Administrator can perform configuration operations at a time. Therefore the role is broken into Full Administrator in config mode and Full Administrator in non-config mode. When a Full Administrator goes into config mode all other Full Administrators will become Full Administrators in non-config mode. Full Administrators in non-config mode cannot configure log settings, network, or cryptography. They retain access to all other functions. All Full Administrators are capable of becoming a Full Administrator in config mode, but only one is authorized at any given time.

The role 'Full Administrator in config mode' fulfills the role of 'Authorized Administrator' defined by the NDPP and can configure cryptographic functionality, configure firewall rules, login banners, cryptographic protocols, update the TOE, and verify updates with a digital signature. Full Administrators can additionally be granted the power of pre-emption. A Full Administrator with pre-emption powers can remove other Full Administrators from config mode. When a user is assigned a Full Administrator role without pre-emption powers they are assigned to either be logged off when pre-empted or to move to non-config mode when pre-empted.
TOE Security Functional Requirements Satisfied: FMT_MTD.1, FMT_SMF.1, FMT_SMR.1.

7.1.6 Protection of the TSF

The TSF protects plaintext passwords, pre-shared keys, symmetric keys, and private keys from being read by hashing prior to storage and enforcing access control on these CSPs. No user passwords can be exported from the TOE.

The TOE provides a reliable time stamp that is used for all TOE functionality, primarily audit log generation. The **System > Time** page of the web management GUI defines the time and date settings. By default, the TOE uses an internal list of public NTP servers to automatically update the time; however in the evaluated configuration, time is set manually, which is achieved by unchecking the "Set time automatically using NTP" and populating the appropriate values for daylight savings time adjustments and time format. Setting the time is restricted to authorized administrators.

Only an authorized administrator can update the TOE. All updates must be digitally signed using the methods described in FCS_COP.1(2). Updates are available at https://www.mysonicwall.com and should be downloaded prior to updating. The SonicWALL server's public key is stored on the appliance as a built-in certificate. During the update the TOE will verify the signature before proceeding with the update. If the signature fails the update will fail. Details on certificates and their use within the TOE can be found in Chapter 10 of *SonicWALL SonicOS v5.9 Administrator's Guide.*

The TSF is also responsible for performing power-up self-tests of cryptographic algorithms and software integrity tests to demonstrate the correct operation of the TSF. The following are the power-up self-tests and their associated error message:

- AES 128, 192, and 256 CBC Known Answer Test (KAT) – NDPP Error State entered with error = (50)
- SHA-1 KAT – FIPS Error State entered with error = (60)
- SHA-256 KAT – NDPP Error State entered with error = (62)
- HMAC-SHA-256 KAT – NDPP Error State entered with error = (72)
- RSA KAT for signature generation and verification – NDPP Error State entered with error = (90)
- DRBG KAT – NDPP Error State entered with error = (20)

The TOE will enter an error state when a self-test fails. No cryptographic operations can take place while in the error state. The TOE must be reset to clear an error state.

The AES KAT encrypts a known plaintext with a known key. It then compares the resultant ciphertext with the expected ciphertext hard-coded in the module. If the two values differ, then the KAT fails. If the two values agree, then the AES KAT decrypts the ciphertext with the known keys and compares the decrypted text with the known plaintext. If they differ, then the test fails. If they are the same, then the test passes.

Both SHA KATs take a message to be hashed. It then computes the hash of the message and compares it to the pre-computed hash. The test fails if they mismatch. The HMAC-SHA-256 test takes a known value and hashes it with a hard-coded HMAC key. The result is then compared to the expected value hard-coded in the module. If the values differ, the test fails. If they agree, the test passes.

For the RSA digital signature verification KAT and RSA digital signature generation KAT, the private key is used to sign a block of data, and the resultant value is compared with the original data. If they are the same, the test fails. If they differ, then the public key is used to verify the ciphertext, and the output is compared to the original data. If they are the same, the test passes. Otherwise, it is failed.

Testing done before the instantiation of a new DRBG includes sending the DRBG instantiation algorithm fixed values of entropy. The output is compared with the value that was expected. If the values match, the test passes. Otherwise, it fails. Error testing is done by forcing an error upon the algorithm. If an algorithm handles the error as expected, the test passes. Otherwise, it fails. The combination of these test ensure that the cryptographic operations of the TOE are performing correctly.

When more than one appliance is used to protect a distributed network, the appliances are connected through an IPsec VPN. Section 7.1.2 describes the IPsec protocol and methods used for this function.
TOE Security Functional Requirements Satisfied: FPT_APW_EXT.1, FPT_ITT.1, FPT_SKP_EXT.1, FPT_STM.1, FPT_TST_EXT.1, FPT_TUD_EXT.1.

7.1.7 TOE Access

All access to the TOE takes place through HTTPS and the TOE web-based management interface. Inactive local and remote sessions to the TOE are terminated after a Security Administrator-configurable time interval between five and 60 minutes. In addition, administrators are provided with the capability to terminate their own session. The Full Administrator in config mode is the Security Administrator. All users are presented with a Security Administrator-configured advisory notice and consent warning message prior to access to the TOE.
TOE Security Functional Requirements Satisfied: FTA_SSL.3, FTA_SSL.4, FTA_SSL_EXT.1, FTA_TAB.1.

7.1.8 Trusted Path/Channels

IPsec VPN tunnels are used to provide a trusted communication channel between the TOE and the external syslog server. This trusted communication prevents disclosure and detects modification. HTTPS is used to provide a trusted path for communications between the TOE and the management console. HTTPS protects these communications from disclosure and detects modification. These management HTTPS sessions are logically distinct from all other TOE communications.
TOE Security Functional Requirements Satisfied: FTP_ITC.1, FTP_TRP.1.

8 Rationale

8.1 Conformance Claims Rationale

This Security Target conforms to Part 2 extended and Part 3 conformant of the Common Criteria Standard for Information Technology Security Evaluations, Version 3.1 Revision 3. This ST conforms to the NDPP v1.1.

8.1.1 Variance between the PP and this ST

In some instances changes were made in this ST from the NDPP. All of these changes are documented below with a rationale for the change.

- An Application Note in the NDPP states that the word "manage" in FMT_MTD.1 is the default requirement for management of TSF data. Other iterations are possible. A table was added to FMT_MTD.1 to include the operations listed in the Application Note for manage and any other operations administrators in the TOE can perform.
- FCS_COP.1(2), FCS_COP.1(3), and FCS_COP.1(4) have been modified from the NDPP text to include the portions of the SFR as written in the CEM that were stricken during the refinement operation. This text has been stricken within the SFR.
- Several SFRs in the NDPP include the word "refinement" to imply that they have been refined. This is redundant, and as a result, the word "refinement" has been removed. SFRs included in this ST have been defined with the appropriate conventions to indicate refinements or other operations, as described in section 6.1.

8.2 Security Requirements Rationale

The following discussion provides detailed evidence of coverage for each security objective.

8.2.1 Rationale for Security Functional Requirements of the TOE Objectives

Table 16 Objectives: SFRs Mapping

Objective	Requirements Addressing the Objective	Rationale
O.PROTECTED_COMMUNICATIONS The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.	FCS_CKM.1 Cryptographic key generation	The requirement meets the objective by ensuring that the TOE can generate cryptographic keys for use during cryptographic operations.
	FCS_CKM_EXT.4 Extended: Cryptographic key destruction	The requirement meets the objective by ensuring that the TOE can zeroize cryptographic keys.
	FCS_COP.1(1) Cryptographic operation (for data encryption/decryption)	The requirement meets the objective by ensuring that the TOE can perform encryption and decryption in accordance with the defined algorithms and key sizes.
	FCS_COP.1(2) Cryptographic operation (for	The requirement meets the objective by ensuring that the

Objective	Requirements Addressing the Objective	Rationale
	cryptographic signature)	TOE can perform cryptographic signature services in accordance with the defined algorithms and key sizes.
	FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)	The requirement meets the objective by ensuring that the TOE can perform cryptographic hashing services in accordance with the defined algorithms and key sizes.
	FCS_COP.1(4) Cryptographic operation (for keyed-hash message authentication)	The requirement meets the objective by ensuring that the TOE can perform cryptographic hashing services in accordance with the defined algorithms and key sizes.
	FCS_HTTPS_EXT.1 Extended: HTTPS	The requirement meets the objective by ensuring that the TOE protects remote communications.
	FCS_IPSEC_EXT.1 Extended: IPSEC	The requirement meets the objective by providing the use of the IPsec protocol to create trusted channels.
	FCS_RBG_EXT.1 Extended: Cryptographic operation (random bit generation)	The requirement meets the objective by ensuring that the TOE can perform random bit generation in accordance with the defined algorithms and key sizes.
	FCS_TLS_EXT.1 Extended: TLS	The requirement meets the objective by ensuring that the TOE protects remote communications.
	FPT_ITT.1 Basic Internal TSF Data Transfer Protection	The requirement meets the objective by ensuring that the TOE provides an encrypted communications channel between appliances. This prevents disclosure of TSF data and detects the modification of the TSF data.
	FPT_SKP_EXT.1 Extended: Management of TSF data (for reading of all symmetric keys)	The requirement meets the objective by ensuring that the TOE prevents reading of all specified cryptographic keys.
	FTP_ITC.1 Inter-TSF trusted channel	The requirement meets the objective by ensuring that the TSF

Objective	Requirements Addressing the Objective	Rationale
		uses an HTTPS/TLS session between itself and the management console.
	FTP_TRP.1 Trusted path	The requirement meets the objective by ensuring that the TOE provides a trusted path between itself and authorized IT entities from disclosure.
O.VERIFIABLE_UPDATES The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the Administrator to be unaltered and (optionally) from a trusted source.	FCS_COP.1(2) Cryptographic operation (for cryptographic signature)	The requirement meets the objective by ensuring that the TOE can perform cryptographic signature services in accordance with the defined algorithms and key sizes.
	FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)	The requirement meets the objective by ensuring that the TOE collects information from the managed machines.
	FPT_TUD_EXT.1 Extended: Trusted update	The requirement meets the objective by ensuring that TOE updates can be verified by an administrator.
O.SYSTEM_MONITORING The TOE will provide the capability to generate audit data and send those data to an external IT entity.	FAU_GEN.1 Audit data generation	The requirement meets this objective by ensuring that the TOE maintains a record of defined security related events and the flow of network traffic, including relevant details about the event.
	FAU_GEN.2 User identity association	The requirement meets the objective by ensuring that the TOE associates each auditable event with the identity of the user that caused the event.
	FAU_STG_EXT.1 Extended: External audit trail storage	The requirement meets the objective by ensuring that the TOE can export audit data to an external syslog server over a secure channel.
	FPT_STM.1 Reliable time stamps	The requirement meets the objective by ensuring that the TOE provides reliable timestamps.
O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	FTA_TAB.1 Default TOE access banners	The requirement meets the objective by ensuring the TSF displays an administrator configurable warning and consent notice to all users at login.

Objective	Requirements Addressing the Objective	Rationale
O.TOE_ADMINISTRATION The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.	FIA_PMG_EXT.1 Extended: Password management	The requirement meets the objective by ensuring that the TOE ensures a user's password meets the defined requirements.
	FIA_UIA_EXT.1 Extended: User identification and authentication	The requirement meets the objective by ensuring that the TOE ensures that a user must be successfully identified and authenticated before being allowed access to TOE management functions.
	FIA_UAU_EXT.2 Extended: Password-based authentication mechanism	The requirement meets the objective by ensuring that the TOE provides a local password based authentication.
	FIA_UAU.7 Protected authentication feedback	The requirement meets the objective by ensuring that the TOE provides obscured feedback while the user is authenticating.
	FMT_MTD.1 Management of TSF data	The requirement meets the objective by ensuring that only authorized users are allowed access to TSF data.
	FMT_SMF.1 Specification of management functions	The requirement meets the objective by ensuring that the TOE includes administrative functions to facilitate the management of the TSF.
	FMT_SMR.2 Restrictions on security roles	The requirement meets the objective by ensuring that the TOE associates users with roles to provide access to TSF management functions and data.
	FPT_APW_EXT.1 Extended: Protection of administrator passwords	The requirement meets the objective by the TOE preventing the reading of plaintext passwords, ensuring that an attacker cannot learn the administrators password.
	FTA_SSL_EXT.1 Extended: TSF-initiated session locking	The requirement meets the objective by ensuring that local administrative sessions are locked out after a period of inactivity. This ensures that an administrator session cannot be hijacked if an administrator workstation is left unattended.

Objective	Requirements Addressing the Objective	Rationale
	FTA_SSL.3 TSF-initiated termination	The requirement meets the objective by ensuring that local administrative sessions are locked out after a period of inactivity. This ensures that an administrator session cannot be hijacked if an administrator workstation is left unattended.
	FTA_SSL.4 User-initiated termination	The requirement meets the objective by ensuring that the TOE provides a mechanism for administrators to terminate active sessions when no longer in use.
O.RESIDUAL_INFORMATION_CLEARING The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.	FDP_RIP.2 Full residual information protection	The requirement meets the objective by ensuring that network packets sent from the TOE do not include "left over" data from the processing of previous network information.
O.SESSION_LOCK The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.	FTA_SSL_EXT.1 Extended: TSF-initiated session locking	The requirement meets the objective by ensuring that the TOE locks the session and requires that the administrator user has to re-authenticate to the TSF prior to unlocking the session.
O.TSF_SELF_TEST The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.	FPT_TST_EXT.1 Extended: TSF testing	The requirement meets the objective by ensuring that the TOE provides some self-tests on a subset of its security functionality to ensure it is operating properly.

8.2.2 Security Assurance Requirements Rationale

The assurance requirements from NDPP v1.1 were chosen to provide independent assurance that due care has been exercised with respect to the protection of personal or similar information. As such, minimal additional tasks are placed upon the vendor assuming the vendor follows reasonable software engineering practices and the TOE functions in a manner consistent with its documentation. The chosen assurance level is appropriate with the threats defined for the environment. The TOE is expected to be in a non-hostile position and embedded in or protected by other products designed to address threats that correspond with the intended environment. At this assurance level, the TOE will have incurred a search for obvious flaws to support its introduction into the non-hostile environment. The augmentation of ALC_FLR.2, ADV_FSP.2, ADV_TDS.1, ATE_FUN.1, ASE_SPD.1, ASE_REQ.2, and ATE_COV.1 were chosen to give greater assurance of the developer's on-going flaw remediation processes, design details, and testing of requirements.

8.2.3 Dependency Rationale

This ST does satisfy all the requirement dependencies of the Common Criteria. Table 17 lists each requirement to which the TOE claims conformance with a dependency and indicates whether the dependent requirement was included. As the table indicates, all dependencies have been met.

Table 17 Functional Requirements Dependencies

SFR ID	Dependencies	Dependency Met	Rationale
FAU_GEN.1	FPT_STM.1	✓	
FAU_GEN.2	FAU_GEN.1	✓	
	FIA_UID.1	✓	Although FIA_UID.1 is not included, FIA_UIA_EXT.1 provides coverage for user identification and authentication which supersedes FIA_UID.1.
FAU_STG_EXT.1	FAU_GEN.1	✓	
	FTP_ITC.1	✓	
FCS_CKM.1	FCS_CKM.4	✓	Although FCS_CKM.4 is not in the ST, FCS_CKM_EXT.4 provides coverage.
	FCS_COP.1(1)	✓	
	FCS_COP.1(2)	✓	
	FCS_COP.1(3)	✓	
	FCS_COP.1(4)	✓	
FCS_CKM_EXT.4	FCS_CKM.1	✓	
FCS_COP.1(1)	FCS_CKM.1	✓	
	FCS_CKM.4	✓	Although FCS_CKM.4 is not in the ST, FCS_CKM_EXT.4 provides coverage.
FCS_COP.1(2)	FCS_CKM.1	✓	
	FCS_CKM.4	✓	Although FCS_CKM.4 is not in the ST, FCS_CKM_EXT.4 provides coverage.
FCS_COP.1(3)	FCS_CKM.1	✓	
	FCS_CKM.4	✓	Although FCS_CKM.4 is not in the ST, FCS_CKM_EXT.4 provides coverage.
FCS_COP.1(4)	FCS_CKM.1	✓	

SFR ID	Dependencies	Dependency Met	Rationale
	FCS_CKM.4	✓	Although FCS_CKM.4 is not in the ST, FCS_CKM_EXT.4 provides coverage.
FCS_HTTPS_EXT.1	FCS_TLS_EXT.1	✓	
FCS_IPSEC_EXT.1	FCS_COP.1(4)	✓	
	FCS_COP.1(2)	✓	
	FCS_COP.1(3)	✓	
FCS_RBG_EXT.1	No dependencies	✓	
FCS_TLS_EXT.1	FCS_COP.1(1)	✓	
FDP_RIP.2	No dependencies	✓	
FIA_PMG_EXT.1	No dependencies	✓	
FIA_UIA_EXT.1	No dependencies	✓	
FIA_UAU_EXT.2	No dependencies	✓	
FIA_UAU.7	FIA_UAU.1	✓	Although FIA_UAU.1 is not included, FIA_UIA_EXT.1 provides coverage for user identification and authentication which supersedes FIA_UAU.1.
FMT_MTD.1	FMT_SMF.1	✓	
	FMT_SMR.1	✓	Although FMT_SMR.1 is not included, FMT_SMR.2 is heirarchical to FMT_SMR.1 and therefore provides the same coverage.
FMT_SMF.1	No dependencies	✓	
FMT_SMR.2	FIA_UID.1	✓	Although FIA_UID.1 is not included, FIA_UIA_EXT.1 provides coverage for user identification and authentication which supersedes FIA_UID.1.
FPT_APW_EXT.1	No dependencies	✓	
FPT_SKP_EXT.1	No dependencies	✓	
FPT_ITT.1	FCS_IPSEC_EXT.1	✓	
FPT_STM.1	No dependencies	✓	

SFR ID	Dependencies	Dependency Met	Rationale
FPT_TST_EXT.1	No dependencies	✓	
FPT_TUD_EXT.1	FCS_COP.1(2)	✓	
	FCS_COP.1(3)	✓	
FTA_SSL_EXT.1	FIA_UAU.1	✓	Although FIA_UAU.1 is not included, FIA_UIA_EXT.1 provides coverage for user identification and authentication which supersedes FIA_UAU.1.
FTA_SSL.3	No dependencies	✓	
FTA_SSL.4	No dependencies	✓	
FTA_TAB.1	No dependencies	✓	
FTP_ITC.1	No dependencies	✓	
FTP_TRP.1	No dependencies	✓	

9 Acronyms and Terms

This section describes the acronyms and terms.

9.1 Terminology

Table 18 Terms

Name	Definition
Authorized Administrator	A user with administrator TOE access that has been successfully identified and authenticated by the TOE.
Config mode	Administrators to can access the TOE in config mode or non-config mode. Configuration changes to the TOE are only authorized in config mode. Only one administrator is authorized to be in config mode at a time to ensure that conflicting changes to TOE configuration are not made simultaneously.
Hardware-based noise source	A hardware random number generator is an apparatus that generates random numbers from a physical process. Such devices are often based on microscopic phenomena that generate a low-level, statistically random "noise" signal, such as thermal noise or the photoelectric effect or other quantum phenomena. These processes are, in theory, completely unpredictable, and the theory's assertions of unpredictability are subject to experimental test. A hardware random number generator typically consists of a transducer to convert some aspect of the physical phenomena to an electrical signal, an amplifier and other electronic circuitry to increase the amplitude of the random fluctuations to a macroscopic level, and some type of analog to digital converter to convert the output into a digital number, often a simple binary digit 0 or 1. By repeatedly sampling the randomly varying signal, a series of random numbers is obtained.
Target network	The domain of network and managed devices to be analyzed by the TOE.

9.2 Acronyms

Table 19 Acronyms

Acronym	Definition
3G	Third Generation
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
ARP	Address Resolution Protocol
CA	Certificate Authority
CAC	Common Access Card
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining

Acronym	Definition
CC	Common Criteria
CEM	Common Evaluation Methodology
CLI	Command Line Interface
CSP	Critical Security Parameters
CTR	Counter Mode
DH	Diffie-Hellman
DHCP	Dynamic Host Configuration Protocol
DMZ	Demilitarized Zone
DNS	Domain Name System
DPI	Deep-packet inspection
DRGB	Deterministic Random Bit Generator
DSA	Digital Signature Algorithm
EAL	Evaluation Assurance Level
EC	Elliptical Curve
ECDSA	Elliptic Curve Digital Signature Algorithm
ECP	Elliptic Curve modulo a Prime
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standard
FTP	File Transfer Protocol
GAV	Gateway Anti-Virus
GCM	Galois/Counter Mode
GUI	Graphical User Interface
HMAC	Hashed Message Authentication Code
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IKE	Internet Key Exchange
IP	Internet Protocol
IPS	Intrusion Prevention Systems
IPsec	Internet Protocol Security
IT	Information Technology
L2B	Layer-2 Bridge
LAN	Local Area Network
MB	Megabyte

Acronym	Definition
MODP	Exponentiation Modulo a Prime
NAT	Network Address Translation
NDPP	Network Device Protection Protocol
NIST	Institute of Standards and Technology
NTP	Network Time Protocol
OS	Operating System
POS	Prime Order Subgroup
PP	Protection Profile
RBG	Random Bit Generation
rDSA	RSA Digital Signature Algorithm
RFC	Request for Comment
SA	Security Association
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SIP	Session Initiated Protocol
SMTP	Simple Mail Transfer Protocol
SPY	Gateway Anti-Spyware
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TCP	Transmission Control Protocol
TOE	Target of Evaluation
TSF	TOE Security Functionality
UDP	User Datagram Protocol
UTM	Unified Threat Management
VoIP	Voice over IP
VPN	Virtual Private Network
WAN	Wide Area Network

Prepared by:
Corsec Security, Inc.

13135 Lee Jackson Memorial Highway
Suite 220
Fairfax, VA 22033
United States of America

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