WatchGuard Fireware OS v12.6.2 on Firebox NGFWs (NDcPP21/STFFW13/VPNGW10) Security Target

Version 0.4
10/01/2020

Prepared for:

WatchGuard

505 Fifth Avenue South, Suite 500, Seattle, WA 98104

Prepared By:

Gossamer Laboratories

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is WatchGuard Fireware OS v12.6.2 on Firebox NGFWs provided by WatchGuard. The TOE is being evaluated as a network infrastructure device performing firewall and VPN.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a parenthetical number placed at the end of the component. For example FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – WatchGuard Fireware OS v12.6.2 on Firebox NGFWs (NDcPP21/STFFW13/VPNGW10) Security Target

ST Version – Version 0.4

ST Date – 10/01/2020

1.2 TOE Reference

TOE Identification – WatchGuard WatchGuard Fireware OS v12.6.2 on Firebox NGFWs

TOE Developer – WatchGuard

Evaluation Sponsor – WatchGuard

1.3 TOE Overview

The Target of Evaluation (TOE) is WatchGuard Fireware OS v12.6.2 on Firebox NGFWs.

The Target of Evaluation (TOE) is WatchGuard's Firebox Next Generation Firewall (NGFW) appliances running FirewareOS version 12.6.2. The product family is a standalone appliance that can be administered locally or remotely without any management software or devices. The product provides controlled connectivity between two or more network environments. It mediates information flows between clients and servers located on internal and external networks governed by the firewalls.

1.4 TOE Description

The WatchGuard firewall appliances provide a broad range of services, features, and capabilities. This ST makes a set of claims regarding the product's security functionality, in the context of an evaluated configuration. The claimed security functionality is a subset of the product's full functionality. The evaluated configuration is a subset of the possible configurations of the product, established according to the evaluated configuration guidance. This part of the ST describes the physical and logical scope and boundaries of the Target of Evaluation (TOE).

1.4.1 TOE Architecture

The TOE is a suite of hardware devices that provide all-in-one network and content security solutions. These devices (known as Firebox Security Appliances) are equipped with a WatchGuard proprietary operating system (OS) called Fireware v12.6.2. Most platform variants of the TOE run different images, however some families of the TOE run on the same image. Further information on the variations of the TOE included in this evaluation can be found in Section 7.

Firebox appliances (running the Fireware OS) separate the organization’s internal networks from external network connections to decrease the risk of an external attack. It protects the internal, private networks from unauthorized users on the Internet. Traffic that enters and leaves the protected networks is examined by the Firebox appliances. They use access policies to identify and filter different types of information and can also control which policies or ports the protected computers can use on the Internet (outbound access).

1.4.1.1 Physical Boundaries

The TOE is a software and hardware TOE. It is a combination of a particular model Firebox appliance and the Fireware v12.6.2 OS software based on Linux Kernel 4.14.83. The table in Section 7 lists all the instances of the TOE that operate in the CC-evaluated configuration mode along with their CPUs and Ethernet Controllers. All listed TOE instances offer the same core functionalities.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by Firebox NGFWs:

- Security audit
- Cryptographic support
- User data protection
- Stateful Traffic Filtering Firewall
- Identification and authentication
- Security management
- Packet Filtering
- Protection of the TSF
- TOE access
- Trusted path/channels

1.4.1.2.1 Security audit

The TOE generates audit logs and has the capability to store them internally and can configure the TOE to send them to an external audit server. The connection between the TOE and the remote audit server is protected with IPsec. The TOE has a disk cleanup procedure where it removes old audit logs to allow space for new ones. When disk space falls below a predefined threshold, the TOE deletes the oldest set of records so it can continue collecting audit records.

1.4.1.2.2 Cryptographic support

The TOE depends on CAVP certified cryptographic algorithms as a part of the WatchGuard Crypto module. Additionally, certain platforms rely on CAVP certified cryptographic algorithms used for hardware acceleration. The TOE protects the confidentiality and integrity of all information as it passes between the TOE and the remote management workstation (via TLS) and also when it passes between the TOE and the local management workstation (via a private, direct serial connection). The TOE achieves this by using validated cryptographic algorithms to perform encryption and the decryption of data according to the TLS protocol. Additionally, all communications with external IT entities are similarly protected using the IPsec protocol.

1.4.1.2.3 User Data Protection

The TOE ensures that residual information is protected from potential reuse in accessible objects such as network packets.

1.4.1.2.4 Stateful Traffic Filtering Firewall

The TOE supports many protocols for packet filtering including icmpv4, icmpv6, ipv4, ipv6, tcp and udp. The firewall rules implement the SPD rules (permit, deny, bypass). Each rule can be configured to log status of packets pertaining to the rule. All codes under each protocol are implemented. The TOE supports FTP for stateful filtering.

Routed packets are forwarded to a TOE interface with the interface's MAC address as the layer-2 destination address. The TOE routes the packets using the presumed destination address in the IP header, in accordance with route tables maintained by the TOE.

IP packets are processed by the WatchGuard's FirewareOS software, which associates them with application-level connections, using the IP packet header fields: source and destination IP address and port, as well as IP protocol. Fragmented packets are reassembled before they are processed.

The TOE mediates the information flows according to an administrator-defined policy. Some of the traffic may be either silently dropped or rejected (with notification to the presumed source).

The TOE's firewall and VPN capabilities are controlled by defining an ordered set of rules in the Security Rule Base. The Rule Base specifies what communication will be allowed to pass and what will be blocked. It specifies the source and destination of the communication, what services can be used, at what times, and whether to log the connection.

1.4.1.2.5 Identification and authentication

The TOE authenticates all administrative users. The TOE requires that users associated with these accounts be identified and authenticated before permitted access to the TOE and TOE security functions. Users may authenticate using either local password authentication or remote password authentication.

1.4.1.2.6 Security management

The TOE provides local management capabilities via a local serial connection and remote management capabilities via Web-Based GUI (TLS/HTTPS). Management functions allow the administrators to configure users, roles, and security policy attributes.

1.4.1.2.7 Packet Filtering

Please see Section 1.4.1.2.4 Stateful Traffic Filtering Firewall for a description of the TOE's packet filtering mechanism.

1.4.1.2.8 Protection of the TSF

The TOE includes capabilities to protect itself from unwanted modification as well as protecting its persistent data.

The TOE does not store passwords in plaintext; they are obfuscated. The TOE does not support any command line capability to view any cryptographic keys generated or used by the TOE.

The TOE provides a timestamp for use with audit records, timing elements of cryptographic functions, and inactivity timeouts. The operating system clock inside the TOE can be used to provide time information, or the TOE can be configured to rely on up to three NTP servers for its time.

The TOE only allows updates after their signature is successfully verified. The TOE update mechanism uses ECDSA with SHA-512 and P-521 to verify the signature of the update package.

The TOE's FIPS executables are signed using ECDSA with SHA-512 and P-521. For all other executables a hash is computed during system installation and configuration and during updates.

During power-up the integrity of all executables is verified. If an integrity test fails in the cryptographic module, the system display an error message and pause boot, requiring a power cycle to restart the device. Also, during power-up, algorithms are tested in the kernel and user-space. If any of these test fail, the TOE is not operational for users.

The TOE protects all communications outside of the TOE with an approved connection method. Administrative configuration is protected by HTTPS/TLS while NTP and Audit Server communications are protected by IPsec.

1.4.1.2.9 TOE access

The TOE can be configured to display a message of the day banner when an administrator establishes an interactive session and subsequently will enforce an administrator-defined inactivity timeout value after which the inactive session (local or remote) will be terminated.

1.4.1.2.10 Trusted path/channels

The TOE protects all communications outside of the physical boundary of the TOE. The TOE utilizes HTTPS/TLS for administrative configuration, while using IPsec to protect communications with Audit and NTP servers.

1.4.2 TOE Documentation

- Firebox Common Criteria Deployment Guide Fireware v12.6.2, Version 1.0, 10 August 2020 (**Admin Guide [AGD]**)

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017.
    - Part 3 Conformant
- Package Claims:
    - PP-Configuration for Network Devices, Stateful Traffic Filter Firewalls, and Virtual Private Network (VPN) Gateways, Version 1.0, 6 March 2020 (CFG_NDcPP-FW-VPNGW_V1.0)
        - The PP-Configuration includes the following components:
            - Base-PP: Collaborative Protection Profile for Network Devices, Version 2.1, 11 March 2019 (NDcPP21)
            - PP-Module for Stateful Traffic Filter Firewalls, Version 1.3, 27 September 2019 (STFFW13)
            - PP-Module for Virtual Private Network (VPN) Gateways, Version 1.0, 17 September 2019 (VPNGW10)
- NIAP Technical Decisions:

Technical Decision	Applied?
0538 – NIT Technical Decision for Outdated link to allowed-with list	Yes
0536 – NIT Technical Decision for Update Verification Inconsistency	Yes
0535 – NIT Technical Decision for Clarification about digital signature algorithms for FTP_TUD.1	Yes
0534 – NIT Technical Decision for Firewall IPv4 & IPv6 testing by default	Yes
0533 – NIT Technical Decision for FTP_ITC.1 with signed downloads	Yes
0532 – NIT Technical Decision for Use of seeds with higher entropy	Yes
0531 – NIT Technical Decision for Challenge-Response for Authentication	No
0530 – NIT Technical Decision for FCS_TLSC_EXT.1.1 5e test clarification	No
0529 – NIT Technical Decision for OCSP and Authority Information Access extension	Yes
0528 – NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	Yes
0520 – VPN Gateway SFR Rationale	Yes
0511 – VPN GW Conformance Claim to allow for a PP-Module	Yes
0484 – NIT Technical Decision for Interactive sessions in FTA_SSL_EXT.1 & FTA_SSL.3	Yes
0483 – NIT Technical Decision for Applicability of FPT_APW_EXT.1	Yes
0482 – NIT Technical Decision for Identification of usage of cryptographic schemes	Yes
0481 – NIT Technical Decision for FCS_(D)TLSC_EXT.X.2 IP addresses in reference identifiers	No
0480 – NIT Technical Decision for Granularity of audit events	Yes
0478 – NIT Technical Decision for Application Notes for FIA_X509_EXT.1 iterations	Yes
0477 – NIT Technical Decision for Clarifying FPT_TUD_EXT.1 Trusted Update	Yes
0475 – NIT Technical Decision for Separate traffic consideration for SSH rekey	No

0453 – NIT Technical Decision for Clarify authentication methods SSH clients can use to authenticate SSH se	No
0451 – NIT Technical Decision for ITT Comm UUID Reference Identifier	No
0450 – NIT Technical Decision for RSA-based ciphers and the Server Key Exchange message	Yes
0447 – NIT Technical Decision for Using 'diffie-hellman-group-exchange-sha256' in FCS_SSHC/S_EXT.1.7	No
0425 – NIT Technical Decision for Cut-and-paste Error for Guidance AA	Yes
0424 – NIT Technical Decision for NDcPP v2.1 Clarification - FCS_SSHC/S_EXT1.5	No
0423 – NIT Technical Decision for Clarification about application of RfI#201726rev2	Yes
0412 – NIT Technical Decision for FCS_SSHS_EXT.1.5 SFR and AA discrepancy	No
0411 – NIT Technical Decision for FCS_SSHC_EXT.1.5, Test 1 - Server and client side seem to be confused	No
0410 – NIT technical decision for Redundant assurance activities associated with FAU_GEN.1	Yes
0409 – NIT decision for Applicability of FIA_AFL.1 to key-based SSH authentication	No
0408 – NIT Technical Decision for local vs. remote administrator accounts	Yes
0407 – NIT Technical Decision for handling Certification of Cloud Deployments	No
0402 – NIT Technical Decision for RSA-based FCS_CKM.2 Selection	Yes
0401 – NIT Technical Decision for Reliance on external servers to meet SFRs	Yes
0400 – NIT Technical Decision for FCS_CKM.2 and elliptic curve-based key establishment	Yes
0399 – NIT Technical Decision for Manual installation of CRL (FIA_X509_EXT.2)	Yes
0398 – NIT Technical Decision for FCS_SSH*EXT.1.1 RFCs for AES-CTR	No
0397 – NIT Technical Decision for Fixing AES-CTR Mode Tests	Yes
0396 – NIT Technical Decision for FCS_TLSC_EXT.1.1, Test 2	No
0395 – NIT Technical Decision for Different Handling of TLS1.1 and TLS1.2	No

2.1 Conformance Rationale

The ST conforms to the NDcPP21 with STFFW13 and VPNGW10 modules. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the NDcPP21/STFFW13/VPNGW10 and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The NDcPP21 Protection Profile and STFFW13 and VPNGW10 Modules offer additional information about the identified security objectives, but that has not been reproduced here and the NDcPP21/STFFW13/VPNGW10 should be consulted if there is interest in that material.

In general, the NDcPP21 has defined Security Objectives appropriate for network infrastructure device performing firewall and VPN and as such are applicable to the WatchGuard Fireware OS v12.6.2 on Firebox NGFWs TOE.

3.1 Security Objectives for the Operational Environment

OE.ADMIN_CREDENTIALS_SECURE The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

OE.COMPONENTS_RUNNING (applies to distributed TOEs only) For distributed TOEs the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.

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

OE.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

OE.NO_THRU_TRAFFIC_PROTECTION The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

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

OE.RESIDUAL_INFORMATION The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.
For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.

OE.UPDATES The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDcPP21 and STFFW13/VPNGW10 modules. The NDcPP21/ STFFW13/VPNGW10 define the following extended requirements and since they are not redefined in this ST, NDcPP21/STFFW13/VPNGW10 should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- NDcPP21:FAU_STG_EXT.1: Protected Audit Event Storage
- NDcPP21:FCS_HTTPS_EXT.1: HTTPS Protocol
- NDcPP21:FCS_IPSEC_EXT.1: IPsec Protocol
- VPNGW10:FCS_IPSEC_EXT.1: Internet Protocol Security (IPsec) Communications
- NDcPP21:FCS_NTP_EXT.1: NTP Protocol
- NDcPP21:FCS_RBG_EXT.1: Random Bit Generation
- NDcPP21:FCS_TLSS_EXT.1: TLS Server Protocol
- STFFW13:FFW_RUL_EXT.1: Stateful Traffic Filtering
- STFFW13:FFW_RUL_EXT.2: Sateful Filtering of Dynamic Protocols
- NDcPP21:FIA_PMG_EXT.1: Password Management
- VPNGW10:FIA_PSK_EXT.1: Pre-Shared Key Composition
- NDcPP21:FIA_UAU_EXT.2: Password-based Authentication Mechanism
- NDcPP21:FIA_UIA_EXT.1: User Identification and Authentication
- NDcPP21:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
- NDcPP21:FIA_X509_EXT.2: X.509 Certificate Authentication
- VPNGW10:FIA_X509_EXT.2: X.509 Certificate Authentication
- NDcPP21:FIA_X509_EXT.3: X.509 Certificate Requests
- VPNGW10:FIA_X509_EXT.3: X.509 Certificate Requests
- VPNGW10:FPF_RUL_EXT.1: Rules for Packet Filtering
- NDcPP21:FPT_APW_EXT.1: Protection of Administrator Passwords
- NDcPP21:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
- NDcPP21:FPT_STM_EXT.1: Reliable Time Stamps
- NDcPP21:FPT_TST_EXT.1: TSF testing
- VPNGW10:FPT_TST_EXT.1: TSF testing
- VPNGW10:FPT_TST_EXT.3: TSF Self-Test with Defined Methods
- NDcPP21:FPT_TUD_EXT.1: Trusted update
- VPNGW10:FPT_TUD_EXT.1: Trusted update
- NDcPP21:FTA_SSL_EXT.1: TSF-initiated Session Locking

5. Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs have all been drawn from the NDcPP21/STFFW13/VPNGW10. The refinements and operations already performed in the NDcPP21/STFFW13/VPNGW10 are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP21/STFFW13/VPNGW10 and any residual operations have been completed herein. Of particular note, the NDcPP21/STFFW13/VPNGW10 made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the NDcPP21/STFFW13/VPNGW10 which includes all the SARs for EAL 1. However, the SARs are effectively refined since requirement-specific 'Assurance Activities' are defined in the NDcPP21/STFFW13/VPNGW10 that serve to ensure corresponding evaluations will yield more practical and consistent assurance than the EAL 1 assurance requirements alone. The NDcPP21/STFFW13/VPNGW10 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by WatchGuard Fireware OS v12.6.2 on Firebox NGFWs TOE.

Requirement Class	Requirement Component
FAU: Security audit	NDcPP21/STFFW13/VPNGW10:FAU_GEN.1: Audit Data Generation
	NDcPP21:FAU_GEN.2: User identity association
	NDcPP21:FAU_STG.1: Protected audit trail storage
	NDcPP21:FAU_STG.3/LocSpace: Action in case of possible audit data loss
	NDcPP21:FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	NDcPP21:FCS_CKM.1: Cryptographic Key Generation
	NDcPP21:FCS_CKM.2: Cryptographic Key Establishment
	NDcPP21:FCS_CKM.4: Cryptographic Key Destruction
	VPNGW10:FCS_CKM.1/IKE: Cryptographic Key Generation (for IKE Peer Authentication)
	NDcPP21:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	VPNGW10:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	NDcPP21:FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	NDcPP21:FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	NDcPP21:FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	NDcPP21:FCS_HTTPS_EXT.1: HTTPS Protocol
	NDcPP21:FCS_IPSEC_EXT.1: IPsec Protocol
	VPNGW10:FCS_IPSEC_EXT.1: Internet Protocol Security (IPsec) Communications
	NDcPP21:FCS_NTP_EXT.1: NTP Protocol
	NDcPP21:FCS_RBG_EXT.1: Random Bit Generation
	NDcPP21:FCS_TLSS_EXT.1: TLS Server Protocol

FDP: User Data Protection	STFFW13:FDP_RIP.2 Full Residual Information Protection
FFW: Stateful Traffic Filtering Firewall	STFFW13:FFW_RUL_EXT.1: Stateful Traffic Filtering
	STFFW13:FFW_RUL_EXT.2: Sateful Filtering of Dynamic Protocols
FIA: Identification and authentication	NDcPP21:FIA_AFL.1: Authentication Failure Management
	NDcPP21:FIA_PMG_EXT.1: Password Management
	VPNGW10:FIA_PSK_EXT.1: Pre-Shared Key Composition
	NDcPP21:FIA_UAU.7: Protected Authentication Feedback
	NDcPP21:FIA_UAU_EXT.2: Password-based Authentication Mechanism
	NDcPP21:FIA_UIA_EXT.1: User Identification and Authentication
	NDcPP21:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	NDcPP21:FIA_X509_EXT.2: X.509 Certificate Authentication
	VPNGW10:FIA_X509_EXT.2: X.509 Certificate Authentication
	NDcPP21:FIA_X509_EXT.3: X.509 Certificate Requests
	VPNGW10:FIA_X509_EXT.3: X.509 Certificate Requests
FMT: Security management	NDcPP21:FMT_MOF.1/Functions: Management of security functions behaviour
	NDcPP21:FMT_MOF.1/ManualUpdate: Management of security functions behaviour
	NDcPP21:FMT_MTD.1/CoreData: Management of TSF Data
	NDcPP21:FMT_MTD.1/CryptoKeys: Management of TSF data
	VPNGW10:FMT_MTD.1/CryptoKeys: Management of TSF data
	NDcPP21:FMT_SMF.1: Specification of Management Functions
	STFFW13:FMT_SMF.1/FFW: Specification of Management Functions
	VPNGW10:FMT_SMF.1: Specification of Management Functions
	NDcPP21:FMT_SMR.2: Restrictions on Security Roles
FPF: Packet Filtering	VPNGW10:FPF_RUL_EXT.1 Rules for Packet Filtering
FPT: Protection of the TSF	NDcPP21:FPT_APW_EXT.1: Protection of Administrator Passwords
	VPNGW10:FPT_FLS.1/SelfTest: Fail Secure (Self-Test Failures)
	NDcPP21:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	NDcPP21:FPT_STM_EXT.1: Reliable Time Stamps
	NDcPP21:FPT_TST_EXT.1: TSF testing
	VPNGW10:FPT_TST_EXT.1: TSF testing
	VPNGW10:FPT_TST_EXT.3: TSF Self-Test with Defined Methods
	NDcPP21:FPT_TUD_EXT.1: Trusted update
	VPNGW10:FPT_TUD_EXT.1: Trusted update
FTA: TOE access	NDcPP21:FTA_SSL.3: TSF-initiated Termination
	NDcPP21:FTA_SSL.4: User-initiated Termination
	NDcPP21:FTA_SSL_EXT.1: TSF-initiated Session Locking
	NDcPP21:FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	NDcPP21:FTP_ITC.1: Inter-TSF trusted channel
	VPNGW10:FTP_ITC.1/VPN: Inter-TSF Trusted Channel (VPN Communications)
	NDcPP21:FTP_TRP.1/Admin: Trusted Path

Table 1 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (NDcPP21/STFFW13/VPNGW10:FAU_GEN.1)

NDcPP21/STFFW13/VPNGW10:FAU_GEN.1.1

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

a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions comprising:
- Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).
- Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
- Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
- Resetting passwords (name of related user account shall be logged).
- [
  - ***Dynamical definition of a rule***
  - ***Establishment of a session***
  - ***Result (i.e., drop, allow) of applying a rule in the ruleset to a network packet***
  - ***Configuration of the ruleset***];

d) Specifically defined auditable events listed in Table 2.

Requirement	Auditable Events	Additional Content
NDcPP21/STFFW13/VPNGW10:FAU_GEN.1	None	None
NDcPP21:FAU_GEN.2	None	None
NDcPP21:FAU_STG.1	None	None
NDcPP21:FAU_STG.3/LocSpace	Low storage space for audit events.	None
NDcPP21:FAU_STG_EXT.1	None	None
NDcPP21:FCS_CKM.1	None	None
VPNGW10:FCS_CKM.1 /IKE	None	None
NDcPP21:FCS_CKM.2	None	None
NDcPP21:FCS_CKM.4	None	None
NDcPP21:FCS_COP.1/DataEncryption	None	None
VPNGW10:FCS_COP.1/DataEncryption	None	None
NDcPP21:FCS_COP.1/Hash	None	None
NDcPP21:FCS_COP.1/KeyedHash	None	None
NDcPP21:FCS_COP.1/SigGen	None	None
NDcPP21:FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
NDcPP21:FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
VPNGW10:FCS_IPSEC_EXT.1	Session Establishment with peer	Entire packet contents of packets transmitted/received during session establishment
NDcPP21:FCS_NTP_EXT.1	Configuration of a new time server Removal of configured time server	Identity of new/removed time server
NDcPP21:FCS_RBG_EXT.1	None	None

NDcPP21:FCS_TLSS_EXT.1	Failure to establish a TLS Session.	Reason for failure.
STFFW13:FDP_RIP.2	None	None
STFFW13:FFW_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport Layer Protocol TOE Interface
STFFW13:FFW_RUL_EXT.2	Dynamical definition of rule. Establishment of a session	None
NDcPP21:FIA_AFL.1	Unsuccessful login attempt limit is met or exceeded.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_PMG_EXT.1	None	None
VPNGW10:FIA_PSK_EXT.1	None	None
NDcPP21:FIA_UAU.7	None	None
NDcPP21:FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate. Any addition, replacement or removal of trust anchors in the TOE's trust store	Reason for failure of certificate validation Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
NDcPP21:FIA_X509_EXT.2	None	None
VPNGW10:FIA_X509_EXT.2	None	None
NDcPP21:FIA_X509_EXT.3	None	None
VPNGW10:FIA_X509_EXT.3	None	None
NDcPP21:FMT_MOF.1/Functions	Modification of the behavior of the transmission of audit data to an external IT entity.	None
NDcPP21:FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	None
NDcPP21:FMT_MTD.1/CoreData	None	None
NDcPP21:FMT_MTD.1/CryptoKeys	Management of cryptographic keys.	None
VPNGW10:FMT_MTD.1/CryptoKeys	None	None
NDcPP21:FMT_SMF.1	All management activities of TSF data.	None
STFFW13:FMT_SMF.1/FFW	All management activities of TSF data (including creation, modification and deletion of firewall rules).	None
VPNGW10:FMT_SMF.1	None	None

NDcPP21:FMT_SMR.2	None	None
VPNGW10:FPF_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport Layer Protocol
NDcPP21:FPT_APW_EXT.1	None	None
VPNGW10:FPT_FLS.1/SelfTest	None	None
NDcPP21:FPT_SKP_EXT.1	None	None
NDcPP21:FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
NDcPP21:FPT_TST_EXT.1	None	None
VPNGW10:FPT_TST_EXT.1	None	None
VPNGW10:FPT_TST_EXT.3 Defined Methods	None	None
NDcPP21:FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None
VPNGW10:FPT_TUD_EXT.1	None	None
NDcPP21:FPT_TUD_EXT.1	None	None
NDcPP21:FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None
NDcPP21:FTA_SSL.4	The termination of an interactive session.	None
NDcPP21:FTA_SSL_EXT.1	(if 'lock the session' is selected) Any attempts at unlocking of an interactive session. (if 'terminate the session' is selected) The termination of a local session by the session locking mechanism.	None
NDcPP21:FTA_TAB.1	None	None
NDcPP21:FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
VPNGW10:FTP_ITC.1/VPN	None	None
NDcPP21:FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path.	None

	Failure of the trusted path functions.	

NDcPP21:FAU_GEN.1.2

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of Table 2.

5.1.1.2 User identity association (NDcPP21:FAU_GEN.2)

NDcPP21:FAU_GEN.2.1

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

5.1.1.3 Protected audit trail storage (NDcPP21:FAU_STG.1)

NDcPP21:FAU_STG.1.1

The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

NDcPP21:FAU_STG.1.2

The TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail.

5.1.1.4 Action in case of possible audit data loss (NDcPP21:FAU_STG.3/LocSpace)

NDcPP21:FAU_STG.3.1/LocSpace

The TSF shall generate a warning to inform the Administrator if the audit trail exceeds the local audit trail storage capacity.

5.1.1.5 Protected Audit Event Storage (NDcPP21:FAU_STG_EXT.1)

NDcPP21:FAU_STG_EXT.1.1

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

NDcPP21:FAU_STG_EXT.1.2

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

[***TOE shall consist of a single standalone component that stores audit data locally***]

NDcPP21:FAU_STG_EXT.1.3

The TSF shall [***overwrite previous audit records according to the following rule: [delete the oldest log upon new log generation (circular buffer)]***] when the local storage space for audit data is full.

5.1.2 Cryptographic support (FCS)

5.1.2.1 Cryptographic Key Generation (NDcPP21:FCS_CKM.1)

NDcPP21:FCS_CKM.1.1

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

- ***RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.3,***
- ***ECC schemes using 'NIST curves' [P-256, P-384] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4,***
- ***FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.1,***

- ***FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3***].

5.1.2.2 Cryptographic Key Generation (for IKE Peer Authentication) (VPNGW10:FCS_CKM.1/IKE)

VPNGW10:FCS_CKM.1.1/IKE

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

- ***FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes;***
- ***FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-256, P-384 and [no other curves]]***

and [

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

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

5.1.2.3 Cryptographic Key Establishment (NDcPP21:FCS_CKM.2)

NDcPP21:FCS_CKM.2.1

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

- ***RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, 'Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1 (TD0402 applied),***
- ***Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography',***
- ***Finite field-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography',***
- ***Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526, Section 3***]. (TD0402 applied)

5.1.2.4 Cryptographic Key Destruction (NDcPP21:FCS_CKM.4)

NDcPP21:FCS_CKM.4.1

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

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

that meets the following: No Standard.

5.1.2.5 Cryptographic Operation (AES Data Encryption/Decryption) (NDcPP21:FCS_COP.1/DataEncryption)

NDcPP21:FCS_COP.1.1/DataEncryption

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [***CBC, GCM***] mode and cryptographic key sizes [***128 bits, 192 bits, 256***

bits] that meet the following: AES as specified in ISO 18033-3, [***CBC as specified in ISO 10116, GCM as specified in ISO 19772***].
(Refinement: 192-bits for CBC only)

5.1.2.6 Cryptographic Operation (AES Data Encryption/Decryption) (VPNGW10:FCS_COP.1/DataEncryption)

VPNGW10:FCS_COP.1.1/DataEncryption
The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [***CBC, GCM***] and [***no other***] mode and cryptographic key sizes [***128 bits, 192 bits, 256 bits***], and [***no other cryptographic key sizes***] that meet the following: AES as specified in ISO 18033-3, [***CBC as specified in ISO 10116, GCM as specified in ISO 19772***] and [***no other standards***].
(Refinement: 192-bits for CBC only)

5.1.2.7 Cryptographic Operation (Hash Algorithm) (NDcPP21:FCS_COP.1/Hash)

NDcPP21:FCS_COP.1.1/Hash
The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-1, SHA-256, SHA-384, SHA-512***] and message digest sizes [***160, 256, 384, 512***] that meet the following: ISO/IEC 10118-3:2004.

5.1.2.8 Cryptographic Operation (Keyed Hash Algorithm) (NDcPP21:FCS_COP.1/KeyedHash)

NDcPP21:FCS_COP.1.1/KeyedHash
The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***] and cryptographic key sizes [***160, 256, 384, 512***] and message digest sizes [***160, 256, 384, 512***] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'.

5.1.2.9 Cryptographic Operation (Signature Generation and Verification) (NDcPP21:FCS_COP.1/SigGen)

NDcPP21:FCS_COP.1.1/SigGen
The TSF shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [

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

that meet the following: [

- ***For RSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,***
- ***For ECDSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4***].

5.1.2.10 HTTPS Protocol (NDcPP21:FCS_HTTPS_EXT.1)

NDcPP21:FCS_HTTPS_EXT.1.1
The TSF shall implement the HTTPS protocol that complies with RFC 2818.
NDcPP21:FCS_HTTPS_EXT.1.2
The TSF shall implement HTTPS using TLS.
NDcPP21:FCS_HTTPS_EXT.1.3
If a peer certificate is presented, the TSF shall [***request authorization to establish the connection***] if the peer certificate is deemed invalid.

5.1.2.11 IPsec Protocol (NDcPP21:FCS_IPSEC_EXT.1)

NDcPP21:FCS_IPSEC_EXT.1.1
The TSF shall implement the IPsec architecture as specified in RFC 4301.

NDcPP21:FCS_IPSEC_EXT.1.2
The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.

NDcPP21:FCS_IPSEC_EXT.1.3
The TSF shall implement [***tunnel mode***].

NDcPP21:FCS_IPSEC_EXT.1.4
The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [***AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified by RFC 3602)***] together with a Secure Hash Algorithm (SHA)-based HMAC [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***] and [***AES-GCM-128, AES-GCM-256 (specified in RFC 4106)***].

NDcPP21:FCS_IPSEC_EXT.1.5
The TSF shall implement the protocol: [
- ***IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [no other RFCs for extended sequence numbers], and [RFC 4868 for hash functions],***
- ***IKEv2 as defined in RFC 5996 and [with mandatory support for NAT traversal as specified in RFC 5996, section 2.23], and [RFC 4868 for hash functions]***].

NDcPP21:FCS_IPSEC_EXT.1.6
The TSF shall ensure the encrypted payload in the [***IKEv1, IKEv2***] protocol uses the cryptographic algorithms [***AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 5282)) (IKEv2 only for GMC ciphers)***].

NDcPP21:FCS_IPSEC_EXT.1.7
The TSF shall ensure that [
- ***IKEv1 Phase 1 SA lifetimes can be configured by a Security Administrator based on [***
  - ***length of time, where the time values can be configured within [[30 minutes to 24] hours],***
- ***IKEv2 SA lifetimes can be configured by a Security Administrator based on [***
  - ***length of time, where the time values can be configured within [[30 minutes to 24] hours]***].

NDcPP21:FCS_IPSEC_EXT.1.8
The TSF shall ensure that [
- ***IKEv1 Phase 2 lifetimes can be configured by a Security Administrator based on [***
  - ***number of bytes,***
  - ***length of time, where the time values can be configured within [5 minutes to 8] hours]***].
- ***IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [***
  - ***number of bytes,***
  - ***length of time, where the time values can be configured within [5 minutes to 8] hours]***].

NDcPP21:FCS_IPSEC_EXT.1.9
The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ('x' in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [***256, 384***] bits.

NDcPP21:FCS_IPSEC_EXT.1.10
The TSF shall generate nonces used in [***IKEv1***, ***IKEv2***] exchanges of length [***- according to the security strength associated with the negotiated Diffie-Hellman group***].

NDcPP21:FCS_IPSEC_EXT.1.11
The TSF shall ensure that all IKE protocols implement DH Group(s) [***14 (2048-bit MODP), 19 (256-bit Random ECP), 20 (384-bit Random ECP)***].

NDcPP21:FCS_IPSEC_EXT.1.12
The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv1 Phase 1, IKEv2 IKE_SA***] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv1 Phase 2, IKEv2 CHILD_SA***] connection.

NDcPP21:FCS_IPSEC_EXT.1.13
The TSF shall ensure that all IKE protocols perform peer authentication using [***RSA, ECDSA***] that use X.509v3 certificates that conform to RFC 4945 and [***Pre-shared Keys***].

NDcPP21:FCS_IPSEC_EXT.1.14
The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: [***Distinguished Name (DN), SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN, CN: IP address, CN: Fully Qualified Domain Name (FQDN), CN: user FQDN, Distinguished Name (DN)***] and [***no other reference identifier type***].

5.1.2.12 Internet Protocol Security (IPsec) Communications (VPNGW10:FCS_IPSEC_EXT.1)

VPNGW10:FCS_IPSEC_EXT.1.1
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.2
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.3
The TSF shall implement [***tunnel mode***].

VPNGW10:FCS_IPSEC_EXT.1.4
The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [***AES-CBC-128, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 4106)***] and [***AES-CBC-192 (specified in RFC 3602)***] together with a Secure Hash Algorithm (SHA)-based HMAC [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***].

VPNGW10:FCS_IPSEC_EXT.1.5
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.6
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.7
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.8
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.9
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.10
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.11
The TSF shall ensure that IKE protocols implement DH Groups 19 (256-bit Random ECP), 20 (384-bit Random ECP), and [***14 (2048-bit MODP)***].

VPNGW10:FCS_IPSEC_EXT.1.12
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.13
See NDcPP21:FCS_IPSEC_EXT.1

VPNGW10:FCS_IPSEC_EXT.1.14
The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: Distinguished Name (DN), [***SAN: IP address, SAN: Fully***

Qualified Domain Name (FQDN), SAN: user FQDN, CN: IP address, CN: Fully Qualified Domain Name (FQDN), CN: user FQDN, Distinguished Name (DN)].

5.1.2.13 NTP Protocol (NDcPP21:FCS_NTP_EXT.1)

NDcPP21:FCS_NTP_EXT.1.1
The TSF shall use only the following NTP version(s) [***NTP v4 (RFC 5905)***].

NDcPP21:FCS_NTP_EXT.1.2
The TSF shall update its system time using [***- [IPsec] to provide trusted communication between itself and an NTP time source.***].

NDcPP21:FCS_NTP_EXT.1.3
The TSF shall not update NTP timestamp from broadcast and/or multicast addresses

NDcPP21:FCS_NTP_EXT.1.4
The TSF shall support configuration of at least three (3) NTP time sources.

5.1.2.14 Random Bit Generation (NDcPP21:FCS_RBG_EXT.1)

NDcPP21:FCS_RBG_EXT.1.1
The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [***CTR_DRBG (AES)***].

NDcPP21:FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[one] software-based noise source***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.2.15 TLS Server Protocol (NDcPP21:FCS_TLSS_EXT.1)

NDcPP21:FCS_TLSS_EXT.1.1
The TSF shall implement [***TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [
TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,
TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246,
TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246,
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288,
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289].

NDcPP21:FCS_TLSS_EXT.1.2
The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [***none***].

NDcPP21:FCS_TLSS_EXT.1.3
The TSF shall [***generate EC Diffie-Hellman parameters over NIST curves [secp384r1] and no other curves, generate Diffie-Hellman parameters of size [2048 bits]***].

5.1.3 User data protection (FDP)

5.1.3.1 Full Residual Information Protection (STFFW13:FDP_RIP.2)

STFFW13:FDP_RIP.2.1

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

5.1.4 Stateful Traffic Filtering Firewall (FFW)

5.1.4.1 Stateful Traffic Filtering (STFFW13:FFW_RUL_EXT.1)

STFFW13:FFW_RUL_EXT.1.1

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

STFFW13:FFW_RUL_EXT.1.2

The TSF shall allow the definition of Stateful Traffic Filtering rules using the following network protocol fields:

--- ICMPv4
- o Type
- o Code

--- ICMPv6
- o Type
- o Code

--- IPv4
- o Source address
- o Destination Address
- o Transport Layer Protocol

--- IPv6
- o Source address
- o Destination Address
- o Transport Layer Protocol
- o [***no other field***]

--- TCP
- o Source Port
- o Destination Port

--- UDP
- o Source Port
- o Destination Port

--- and distinct interface.

STFFW13:FFW_RUL_EXT.1.3

The TSF shall allow the following operations to be associated with Stateful Traffic Filtering rules: permit or drop with the capability to log the operation.

STFFW13:FFW_RUL_EXT.1.4

The TSF shall allow the Stateful Traffic Filtering rules to be assigned to each distinct network interface.

STFFW13:FFW_RUL_EXT.1.5

The TSF shall:

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

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

STFFW13:FFW_RUL_EXT.1.6

The TSF shall enforce the following default Stateful Traffic Filtering rules on all network traffic:

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

STFFW13:FFW_RUL_EXT.1.7

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

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

STFFW13:FFW_RUL_EXT.1.8

The TSF shall process the applicable Stateful Traffic Filtering rules in an administratively defined order.

STFFW13:FFW_RUL_EXT.1.9

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

STFFW13:FFW_RUL_EXT.1.10

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

5.1.4.2 Sateful Filtering of Dynamic Protocols (STFFW13:FFW_RUL_EXT.2)

STFFW13:FFW_RUL_EXT.2.1

The TSF shall dynamically define rules or establish sessions allowing network traffic to flow for the following network protocols [***FTP***]

5.1.5 Identification and authentication (FIA)

5.1.5.1 Authentication Failure Management (NDcPP21:FIA_AFL.1)

NDcPP21:FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within [***2-200***]

unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password. (TD0408 applied)

NDcPP21:FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met, the TSF shall [***prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until [an unlock action] is taken by an Administrator, prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until an Administrator defined time period has elapsed***]. (TD0408 applied)

5.1.5.2 Password Management (NDcPP21:FIA_PMG_EXT.1)

NDcPP21:FIA_PMG_EXT.1.1

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

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

b) Minimum password length shall be configurable to between [***8***] and [***32***] characters.

5.1.5.3 Pre-Shared Key Composition (VPNGW10:FIA_PSK_EXT.1)

VPNGW10:FIA_PSK_EXT.1.1

The TSF shall be able to use pre-shared keys for IPsec and [***no other protocols***].

VPNGW10:FIA_PSK_EXT.1.2

The TSF shall be able to accept text-based pre-shared keys that:

- Are 22 characters and [***[1-79 characters]***];
- composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”).

VPNGW10:FIA_PSK_EXT.1.3

The TSF shall condition the text-based pre-shared keys by using [***SHA-1, SHA-256, SHA-512, [SHA-384]***].

VPNGW10:FIA_PSK_EXT.1.4

The TSF shall be able to [***accept***] bit-based pre-shared keys.

5.1.5.4 Protected Authentication Feedback (NDcPP21:FIA_UAU.7)

NDcPP21:FIA_UAU.7.1

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

5.1.5.5 Password-based Authentication Mechanism (NDcPP21:FIA_UAU_EXT.2)

NDcPP21:FIA_UAU_EXT.2.1

The TSF shall provide a local [***password-based***] authentication mechanism to perform local administrative user authentication. (TD0408 applied)

5.1.5.6 User Identification and Authentication (NDcPP21:FIA_UIA_EXT.1)

NDcPP21: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;
- [***Allow the user to select authentication server***].

NDcPP21:FIA_UIA_EXT.1.2

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

5.1.5.7 X.509 Certificate Validation (NDcPP21:FIA_X509_EXT.1/Rev)

NDcPP21:FIA_X509_EXT.1.1/Rev

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

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [***the Online Certificate Status Protocol (OCSP) as specified in RFC 6960***]
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
  - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
  - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
  - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

NDcPP21:FIA_X509_EXT.1.2/Rev

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

5.1.5.8 X.509 Certificate Authentication (NDcPP21:FIA_X509_EXT.2)

NDcPP21:FIA_X509_EXT.2.1

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

NDcPP21:FIA_X509_EXT.2.2

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

5.1.5.9 X.509 Certificate Authentication (VPNGW10:FIA_X509_EXT.2)

VPNGW10:FIA_X509_EXT.2.1

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

VPNGW10:FIA_X509_EXT.2.2

See NDcPP21:FIA_X509_EXT.2

5.1.5.10 X.509 Certificate Requests (NDcPP21/VPNGW10:FIA_X509_EXT.3)

NDcPP21/VPNGW10:FIA_X509_EXT.3.1

The TSF shall generate a Certification Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [***Common Name, Organization, Organizational Unit, Country***]. (TD0333 applied)

NDcPP21/VPNGW10:FIA_X509_EXT.3.2

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

5.1.6 Security management (FMT)

5.1.6.1 Management of security functions behaviour (NDcPP21:FMT_MOF.1/Functions)

NDcPP21:FMT_MOF.1.1/Functions

The TSF shall restrict the ability to [***determine the behaviour of, modify the behaviour of***] the functions [***transmission of audit data to an external IT entity***] to Security Administrators.

5.1.6.2 Management of security functions behaviour (NDcPP21:FMT_MOF.1/ManualUpdate)

NDcPP21:FMT_MOF.1.1/ManualUpdate

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

5.1.6.3 Management of TSF Data (NDcPP21:FMT_MTD.1/CoreData)

NDcPP21:FMT_MTD.1.1/CoreData

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

5.1.6.4 Management of TSF data (NDcPP21:FMT_MTD.1/CryptoKeys)

NDcPP21:FMT_MTD.1.1/CryptoKeys

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

5.1.6.5 Management of TSF data (VPNGW10:FMT_MTD.1/CryptoKeys)

VPNGW10:FMT_MTD.1.1/CryptoKeys

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

5.1.6.6 Specification of Management Functions (NDcPP21/VPNGW10:FMT_SMF.1)

NDcPP21/VPNGW10:FMT_SMF.1.1

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

- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using digital signature and [***no other***] capability prior to installing those updates;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- Ability to manage the cryptographic keys;
- Ability to configure the cryptographic functionality;
- Ability to configure the lifetime for IPsec SAs;
- Ability to import X.509v3 certificates to the TOE's trust store;
- Ability to enable, disable, determine and modify the behavior of all the security functions of the TOE identified in this PP-Module;
- Ability to configure all security management functions identified in other sections of this PP-Module;
- [
  - ***Ability to configure audit behavior,***
  - ***Ability to modify the behavior of the transmission of audit data to an external IT entity,***
  - ***Ability to configure the reference identifier for the peer,***
  - ***Ability to configure the lifetime for IPsec SAs,***
  - ***Ability to re-enable an Administrator account,***
  - ***Ability to set the time which is used for time-stamps;***

- ***Ability to configure NTP***
- ***Ability to enable/disable a ruleset on a network interface***
- ***Ability to configure a ruleset***
- ***Ability to configure the supported algorithms***].

5.1.6.7 Specification of Management Functions (STFFW13:FMT_SMF.1/FFW)

STFFW13:FMT_SMF.1.1/FFW

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

- Ability to configure firewall rules;

5.1.6.8 Restrictions on Security Roles (NDcPP21:FMT_SMR.2)

NDcPP21:FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

NDcPP21:FMT_SMR.2.2

The TSF shall be able to associate users with roles.

NDcPP21:FMT_SMR.2.3

The TSF shall ensure that the conditions

- The Security Administrator role shall be able to administer the TOE locally;
- The Security Administrator role shall be able to administer the TOE remotely are satisfied.

5.1.7 Packet Filtering (FPF)

5.1.7.1 Rules for Packet Filtering (VPNGW10:FPF_RUL_EXT.1)

VPNGW10:FPF_RUL_EXT.1.1

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

VPNGW10:FPF_RUL_EXT.1.2

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

- IPv4 (RFC 791)
  - Source address
  - Destination Address
  - Protocol
- IPv6 (RFC 2460)
  - Source address
  - Destination Address
  - Next Header (Protocol)
- TCP (RFC 793)
  - Source Port
  - Destination Port
- UDP (RFC 768)
  - Source Port
  - Destination Port

VPNGW10:FPF_RUL_EXT.1.3

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

VPNGW10:FPF_RUL_EXT.1.4

The TSF shall allow the Packet Traffic Filtering rules to be assigned to each distinct network interface.

VPNGW10:FPF_RUL_EXT.1.5
The TSF shall process the applicable Packet Filtering rules (as determined in accordance with FPF_RUL_EXT.1.4) in the following order: Administrator-defined.
VPNGW10:FPF_RUL_EXT.1.6
The TSF shall drop traffic if a matching rule is not identified.

5.1.8 Protection of the TSF (FPT)

5.1.8.1 Protection of Administrator Passwords (NDcPP21:FPT_APW_EXT.1)

NDcPP21:FPT_APW_EXT.1.1
The TSF shall store administrative passwords in non-plaintext form. (TD0483 applied).
NDcPP21:FPT_APW_EXT.1.2
The TSF shall prevent the reading of plaintext administrative passwords. (TD0483 applied).

5.1.8.2 Fail Secure (Self-Test Failures) (VPNGW10:FPT_FLS.1/SelfTest)

VPNGW10:FPT_FLS.1.1/SelfTest
The TSF shall shut down when the following types of failures occur: failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests.

5.1.8.3 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) (NDcPP21:FPT_SKP_EXT.1)

NDcPP21:FPT_SKP_EXT.1.1
The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.1.8.4 Reliable Time Stamps (NDcPP21:FPT_STM_EXT.1)

NDcPP21:FPT_STM_EXT.1.1
The TSF shall be able to provide reliable time stamps for its own use.
NDcPP21:FPT_STM_EXT.1.2
The TSF shall [***allow the Security Administrator to set the time, synchronise time with an NTP server***].

5.1.8.5 TSF testing (NDcPP21:FPT_TST_EXT.1)

NDcPP21:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [***during initial start-up (on power on)***] to demonstrate the correct operation of the TSF: [***noise source health tests, FIPS 140-2 standard power-up firmware integrity test, cryptographic module known-answer self-tests***].

5.1.8.6 TSF testing (VPNGW10:FPT_TST_EXT.1)

VPNGW10:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [***during initial startup (on power on)***] to demonstrate the correct operation of the TSF: noise source health tests, [***FIPS 140-2 standard power-up firmware integrity test, cryptographic module known-answer self-tests***].

5.1.8.7 TSF Self-Test with Defined Methods (VPNGW10:FPT_TST_EXT.3)

VPNGW10:FPT_TST_EXT.3.1
The TSF shall run a suite of the following self-tests when loaded for execution to demonstrate the correct operation of the TSF: integrity verification of stored executable code.
VPNGW10:FPT_TST_EXT.3.2

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

5.1.8.8 Trusted update (NDcPP21:FPT_TUD_EXT.1)

NDcPP21:FPT_TUD_EXT.1.1
The TSF shall provide Security Administrators the ability to query the currently executing version of the TOE firmware/software and [***no other TOE firmware/software version***].

NDcPP21:FPT_TUD_EXT.1.2
The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [***no other update mechanism***].

NDcPP21:FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature mechanism***] prior to installing those updates.

5.1.8.9 Trusted update (VPNGW10:FPT_TUD_EXT.1)

VPNGW10:FPT_TUD_EXT.1.1
See NDcPP21:FPT_TUD_EXT.1

VPNGW10:FPT_TUD_EXT.1.2
See NDcPP21:FPT_TUD_EXT.1

VPNGW10:FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a digital signature mechanism and [***no other mechanisms***] prior to installing those updates.

5.1.9 TOE access (FTA)

5.1.9.1 TSF-initiated Termination (NDcPP21:FTA_SSL.3)

NDcPP21:FTA_SSL.3.1
The TSF shall terminate a remote interactive session after a Security Administrator-configurable time interval of session inactivity.

5.1.9.2 User-initiated Termination (NDcPP21:FTA_SSL.4)

NDcPP21:FTA_SSL.4.1
The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.1.9.3 TSF-initiated Session Locking (NDcPP21:FTA_SSL_EXT.1)

NDcPP21:FTA_SSL_EXT.1.1
The TSF shall, for local interactive sessions, [
- ***terminate the session***]

after a Security Administrator-specified time period of inactivity.

5.1.9.4 Default TOE Access Banners (NDcPP21:FTA_TAB.1)

NDcPP21:FTA_TAB.1.1
Before establishing an administrative user session the TSF shall display a Security Administrator-specified advisory notice and consent warning message regarding use of the TOE.

5.1.10 Trusted path/channels (FTP)

5.1.10.1 Inter-TSF trusted channel (NDcPP21:FTP_ITC.1)

NDcPP21:FTP_ITC.1.1

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

NDcPP21:FTP_ITC.1.2

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

NDcPP21:FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [***Audit logging, NTP***].

5.1.10.2 Inter-TSF Trusted Channel (VPN Communications) (VPNGW10:FTP_ITC.1/VPN)

VPNGW10:FTP_ITC.1.1/VPN

The TSF shall be capable of using IPsec to provide a communication channel between itself and authorized IT entities supporting VPN communications that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data

VPNGW10:FTP_ITC.1.2/VPN

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

VPNGW10:FTP_ITC.1.3/VPN

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

5.1.10.3 Trusted Path (NDcPP21:FTP_TRP.1/Admin)

NDcPP21:FTP_TRP.1.1/Admin

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

NDcPP21:FTP_TRP.1.2/Admin

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

NDcPP21:FTP_TRP.1.3/Admin

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

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic Functional Specification
AGD: Guidance documents	AGD_OPE.1: Operational User Guidance
	AGD_PRE.1: Preparative Procedures
ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM Coverage
ATE: Tests	ATE_IND.1: Independent Testing - Conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability Survey

Table 2 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic Functional Specification (ADV_FSP.1)

ADV_FSP.1.1d
The developer shall provide a functional specification.

ADV_FSP.1.2d
The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.1.1c
The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2c
The TSF shall support mutual authentication of TLS clients using X.509v3 certificates.

ADV_FSP.1.3c
The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4c
The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e
The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1d
The developer shall provide operational user guidance.

AGD_OPE.1.1c
The operational user guidance shall describe, for each user role, the useraccessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c
The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c
The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c
The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c
The operational user guidance shall identify all possible modes of operation of the TOE (including

operation following failure or operational error), their consequences, and implications for maintaining secure operation.

AGD_OPE.1.6c

The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c

The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d

The developer shall provide the TOE, including its preparative procedures.

AGD_PRE.1.1c

The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c

The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e

The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d

The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c

The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3.2 TOE CM Coverage (ALC_CMS.1)

ALC_CMS.1.1d

The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c

The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2c

The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.4 Tests (ATE)

5.2.4.1 Independent Testing - Conformance (ATE_IND.1)

ATE_IND.1.1d
The developer shall provide the TOE for testing.

ATE_IND.1.1c
The TOE shall be suitable for testing.

ATE_IND.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2e
The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability Survey (AVA_VAN.1)

AVA_VAN.1.1d
The developer shall provide the TOE for testing.

AVA_VAN.1.1c
The TOE shall be suitable for testing.

AVA_VAN.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence..

AVA_VAN.1.2e
The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3e
The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6. TOE Summary Specification

This chapter describes the security functions:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

6.1 Security audit

The TOE generates log files with information about security related events that the Administrator of the TOE can review to monitor the network security and activity, identify security risks, and address them.

A log file is a list of events along with information about those events. An event is a single activity that occurs on the TOE. For example, TOE's denying of a packet based on a policy set is an event. The TOE also captures information about allowed events to give a more complete picture of the activities on the network.

The TOE is designed to produce syslog conformant messages. It generates and saves the following types of log messages (each with its own audit storage size):

Traffic log messages (512kb) – The TOE generates traffic log messages as it applies packet filter and proxy rules to traffic that goes through the device.

Alarm log messages (256kb) – Alarm log messages are sent when an event occurs that triggers the TOE to run a command. When the alarm condition is matched, the device generates an alarm log message, and then it does the specified action.

Event log messages (Admin log messages – 256kb) – The TOE sends event log messages because of user activity. Actions that can cause the TOE to send an event log message are:

- Device start up and shut down
- Device authentication
- Process start up and shut down
- Problems with the device hardware components
- Any task done by the administrator

Debug log messages (256kb) – Debug log messages include diagnostic information that can be used to troubleshoot problems.

Statistic log messages (Not stored locally, sent to remote log server immediately) – Statistic log messages include information about the performance of the TOE.

All audit records include at least the following information:

- The type of event
- The identity of the subject that caused the event
- The outcome of the event
- The date and time of the event.

The above logs are stored internally on the appliance in a secure method, preventing any unauthorized users from viewing or otherwise accessing the audit records.

The Security audit function satisfies the following security functional requirements:

- NDcPP21:STFFW13:FAU_GEN.1: The TOE can generate audit records for events include starting and stopping the audit function, administrator commands, and all other events identified in section 5.1.1. Furthermore, each audit record identifies the date/time, event type, outcome of the event, responsible subject/user, as well as the additional event-specific content indicated in section 5.1.1. For cryptographic keys, the act of importing and deleting a key is audited and the associated administrator account that performed the action is recorded. The TOE additionally logs administrative events and traffic information related to the firewall functionality of the device. Any changes to rulesets are logged with the complete details about the modified ruleset. Any time traffic matches a configured ruleset (that's been configured to log a match), a log will be generated for the event.
- NDcPP21:FAU_GEN.2: The TOE identifies the responsible user for each event based on the specific administrator or network entity (identified by IP address) that caused the event.
- NDcPP21:FAU_STG.1: Audit records are protected from all modification. While the administrator can configure if audits are stored locally or not, which syslog servers are used (if any), and modify audit record levels for multiple categories, there is no option to add, delete, or modify existing audit records via the CLI or WebUI (or any other method). The WebUI does not provide a modification option and the CLI is a proprietary interface that does not provide the option for any administrator to modify audit records.
- NDcPP21:FAU_STG.3/LocSpace: When locally stored audit records fill their allocated disk space, an audit record is generated informing the administrator that old logs will be deleted to make room for new records.
- NDcPP21:FAU_STG_EXT.1: The TOE can be configured to export audit records to an external SYSLOG server. This communication is protected with the use of IPsec. Audit records are sent to SYSLOG the server(s) in real-time as they are generated. Audit records can only be viewed on the TOE by an authorized administrator via the WebUI. The TOE stores audit records locally based on the syslog audit type. Traffic Log storage is 512kb, Alarm Log storage is 256kb, Event Log storage is 256kb, and Debug Log storage is 256kb. Statistic Logs are immediately sent to an external syslog server and not stored locally. If configured, audit records are stored locally in a circular file buffer where the oldest set of audit records are deleted when the allocated size is exceeded. Each time a set of audit records are deleted, an audit is generated informing the administrator that old logs were deleted.

6.2 Cryptographic support

The TOE supports a range of cryptographic services provided by the Firebox Cryptographic Module Version 1.0 (Firmware) in the evaluated TOE models identified in Section 7. The algorithms and certificates listed below are used for all cryptographic functionality on each of the evaluated products except IPsec encryption, decryption, hashing, and keyed hashing.

The following functions have been CAVP tested and represent the CAVP certificates for the Firebox Cryptographic Module v1.0, which is equivalent across all evaluated platforms.

Requirement	Functions	Cert #
	Encryption/Decryption	
FCS_COP.1/DataEncryption	AES CBC AES GCM	C1968
	Cryptographic signature services	
FCS_COP.1/SigGen	RSA Digital Signature Algorithm ECDSA Digital Signature Algorithm	C1968
	Cryptographic hashing	

FCS_COP.1/Hash	SHA	C1968
	Keyed-hash message authentication	
FCS_COP.1/KeyedHash	HMAC-SHA	C1968
	Random bit generation	
FCS_RBG_EXT.1	CTR_DRBG with SW based noise sources with a minimum of 256 bits of non-determinism	C1968
FCS_CKM.1	Key Generation	C1968
	ECDSA Key Generation RSA Key Generation DSA Key Generation	
FCS_CKM.2	Key Establishment	C1968
	KAS ECC KAS FFC	C1968

Table 3 Software Crypto Module CAVP Certificates

The majority of the TOEs (excluding the T55 and T70) contain hardware acceleration modules that are used to accelerate the functionality of IPsec.

As a special case, the M270's hardware acceleration chip does not support SHA-384 and HMAC-SHA-384, but the device was programmed to use the Firebox Cryptographic Module software to provide support for SHA/HMAC-SHA-384. As such, below you will see two certificates for the M270's SHA based functions.

The following functions have been CAVP tested and represent the CAVP certificates for the hardware acceleration processors present on the platforms (where available):

Platform:	Algorithm:	Certificate:
T35	SHA	SHS 4681
	HMAC-SHA	HMAC 3905
	AES-CBC, AES-GCM	AES 5925
T55/70	SHA	N/A – No HW acceleration module
	HMAC-SHA	N/A – No HW acceleration module
	AES-CBC, AES-GCM	N/A – No HW acceleration module
T20	SHA	C1960
	HMAC-SHA	C1960
	AES-CBC, AES-GCM	C1960
T40	SHA	C1960
	HMAC-SHA	C1960
	AES-CBC, AES-GCM	C1960
T80	SHA	C1965
	HMAC-SHA	C1965
	AES-CBC, AES-GCM	C1965

M270	SHA	SHA-256/512: SHS 4677 SHA-384: C1968
	HMAC-SHA	HMAC-SHA-256/512: HMAC 3901 HMAC-SHA-384: C1968
	AES-CBC, AES-GCM	AES 5921
M370	SHA	SHS 4678
	HMAC-SHA	HMAC 3902
	AES-CBC, AES-GCM	AES 5922
M470	SHA	SHS 4678
	HMAC-SHA	HMAC 3902
	AES-CBC, AES-GCM	AES 5922
M570	SHA	SHS 4678
	HMAC-SHA	HMAC 3902
	AES-CBC, AES-GCM	AES 5922
M670	SHA	SHS 4679
	HMAC-SHA	HMAC 3903
	AES-CBC, AES-GCM	AES 5923
M4600	SHA	SHS 4679
	HMAC-SHA	HMAC 3903
	AES-CBC, AES-GCM	AES 5923
M5600	SHA	SHS 4679
	HMAC-SHA	HMAC 3903
	AES-CBC, AES-GCM	AES 5923

Table 4 Hardware Acceleration Modules CAVP Certificates

The TOE provides key generation for asymmetric keys on all components and can generate 2048-bit and larger RSA keys, DSA 2048-bit keys, and ECDSA keys using NIST curve sizes P-256 and P-384 [according to FIPS PUB 186-4]. The TOE supports DH group 14 key establishment scheme that meets standard RFC 3526, section 3 (for interoperability) as well as group 19 (ECP-256) and group 20 (ECP-384).

The TOE uses a software-based random bit generator that complies with Special Publication 800-90 using CTR_DRBG. AES-256 is used in conjunction with a minimum of 256-bits of entropy.

The TOEs implement the IPsec architecture as specified in RFC 4301. SPD rules can be configured using the firewall rules and VPN communities. Firewall rules are used to distinguish between DROP actions and others, while VPN communities distinguish between traffic that is encrypted (PROTECT) and traffic that is not (BYPASS). If traffic is part of a defined VPN, it will be encrypted and then firewall rules will be applied. Rules are explicit, therefore any packet not matching a rule will be dropped. Rules are processed in order with the first matching rule being applied to the traffic. The TOE supports IKEv1 in main mode and IKEv2 in tunnel mode only. Additionally, IKEv2 supports NAT traversal as specified in RFC 5996. The authorized administrator can configure the TOE to support maximum lifetimes for IKEv1 and IKEv2 SAs based on elapsed time and number of bytes transferred. The TOE allows the Administrator to configure the IKEv1 Phase 1 SA/IKEv2 SA and IKEv1 Phase 2 SA/IKEv2 child SA lifetime by

minutes and the TOE additionally allows the Administrator to configure the IKEv1 Phase 2 SA/IKEv2 child SA lifetime by number of bytes.

For IKE SA/Phase 1, IKEv1 and IKEv2 both support RFC 3602 conformant AES-CBC-128, AES-CBC-192, and AES-CBC-256. IKEv2 along provides support for RFC 4106 conformant AES-GCM-128 and AES-GCM-256 in Phase 1. IKEv1 and IKEv2 both support RFC 3602 conformant AES-CBC-128, AES-CBC-192, and AES-CBC-256, and 4106 conformant AES-GCM-128 and AES-GCM-256 as encryption algorithms for Phase 2/ESP. The TOE also implements HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512 as integrity/authentication algorithms as well as Diffie-Hellman Groups 14, 19, and 20. The administrator configures the order the groups will be negotiated with a peer. TOE generates the secret value x used in the IKEv2 Diffie-Hellman key exchange ('x' in g^x mod p) using the CTR_DRBG specified in FCS_RBG_EXT.1 and having possible lengths of 256 or 384 bits. The TOE generates nonces used in the IKEv1/2 exchanges according to the security strength associated with the negotiated Diffie-Hellman group . The TOE verifies that the default strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the IKEv1/2 Phase 1/IKE_SA connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the IKEv1/2 Phase 2/CHILD_SA connection, preventing configuration of a too-large Phase 2/Child algorithm strength.

The TOE's IPsec implementation supports Pre-Shared Keys (PSKs) and X.509 certificates (both RSA and ECDSA) for IKE authentication.

The TOEs use IPsec to protect communications with external IT entities such as an NTP or Syslog server, with NTP version 4 being supported by the TOE. Further, remote administration is protected via an HTTPS/TLS connection.

The TOE supports TLS v1.1 (RFC4346), and TLS v1.2 (RFC 5246) secure communication protocols and rejects all earlier protocol versions. The TOE supports the following ciphersuites when the TOE is a server:

- TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
- TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246,
- TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246,
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288,
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288,
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289,
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

The TOE also supports HTTPS that complies with RFC 2818 which is used for remote TOE administration via a Web GUI. The TOE's TLS/HTTPS server uses RSA 2048+ bits certificate or an ECDHE P-256/P-384/P-521 for TLS authentication. After the TLS session's successful setup, the administrator must log into the TOE via user name and passwords. The Diffie-Hellman group 14 with parameters of size 2048 bits is used for the key agreement

CSP or Key:	Stored in	Zeroized upon:	Zeroized by:
VPN IKE_SA Keys (Auth initiator and responder, Encryption initiator and responder)	Memory	When IKE SA expired	Overwriting with zeros
VPN CHILD/IPSEC_SA Keys (initiator and responder)	Memory	When child or IKE SA expired	Overwriting with zeros
User IPsec X.509v3 Certs (ECDSA) (public)	On Disk	N/A – Public information	N/A – Public information
VPN PSK	On Disk	Never (may be replaced	
Password hash	On Disk	Never (may be replaced)	
TLS host private key	Command	Flash	Overwriting once with zeros
TLS host digital certificate	Command	Flash	Overwriting once with zeros
TLS pre-master secret	Handshake done	RAM	Overwriting once with zeros
TLS session key	Close of session	RAM	Overwriting once with zeros
User Password	Command	Flash	Overwriting once with zeros
DRBG Seed	Every 100ms	RAM	Overwritten with new value

Table 5 CSPs and Keys

The Cryptographic support function satisfies the following security functional requirements:

- NDcPP21:FCS_CKM.1: The TOE supports RSA, DSA, and ECDSA key generation. The key generation is used by the TOE when it creates a Certificate Signing Request (CSR) to apply for a certificate from the Certificate Authority (CA). The TOE enforces 2048-bit+ key generation for RSA, 2048-bit for DSA, and NIST curves P256 and P384 for ECDSA key pairs.
- VPNGW10:FCS_CKM.1/IKE: See NDcPP21:FCS_CKM.1
- NDcPP21:FCS_CKM.2: The TOE acts as both receiver for RSA-based key establishment, as a sender and receiver for Diffie-Hellman based and Elliptic Curve Diffie-Hellman (ECDH) key establishment in cryptographic operations. Additionally, the TOE implementation of Diffie-Hellman-group-14 meets RFC 3526, Section 3 by virtue of using a 2048-bit MODP group for key establishment.

Scheme	SFR	Service
RSA	FCS_TLSS_EXT.1	Administration
Diffie-Hellman	FCS_TLSS_EXT.1	Administration
ECDH	FCS_TLSS_EXT.1	Administration
ECDH	FCS_IPSEC_EXT.1	Syslog and NTP
Diffie-Hellman (Group 14)	FCS_IPSEC_EXT.1	Syslog and NTP

- NDcPP21:FCS_CKM.4: Keys are zeroized when they are no longer needed by the TOE or as identified above.

- NDcPP21:VPNGW10:FCS_COP.1/DataEncryption: The TOE performs encryption and decryption using AES in CBC and GCM mode with key sizes of either 128 or 256. The corresponding CAVP certificates are identified in the table above for each algorithm on each hardware platform.
- NDcPP21:FCS_COP.1/Hash: The TOE supports cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512, with digest sizes 160, 256, 384, and 512. The security hashing functions are used in IPsec IKEv2 and ESP to provide data packet integrity. The corresponding CAVP certificates are identified in the table above.
- NDcPP21:FCS_COP.1/KeyedHash: The TOE supports keyed-hash message authentication using HMAC-SHA-1/256/384/512 using SHA-1/256/384/512 with 160/256/384/512-bit keys to produce a 160/256/384/512 output MAC (all respectively listed). The SHA-1 and SHA-256 algorithm have block sizes of 512-bits while SHA-384 and SHA-512 have a block sizes of 1024. The corresponding CAVP certificates are identified in the table above.
- NDcPP21:FCS_COP.1/SigGen: The TOE supports the use of RSA sizes 2048 and 3072 and ECDSA with curves P-256, P-384, P-521 cryptographic signatures. Digital signatures are used on product updates. The corresponding CAVP certificates are identified in the table above.
- NDcPP21:FCS_HTTPS_EXT.1: The TOE provides a Web GUI for remote administration and fully supports RFC 2818. The TOE acts as an HTTPS server and waits for client connections on TCP port 443. The TOE's HTTPS server supports TLS version 1.1/1.2 only and will deny connection requests from TLS clients with lower or higher versions.
- NDcPP21/VPNGW10:FCS_IPSEC_EXT.1: The TOE supports IPsec when exporting audit logs to an external server, protecting the NTP server connection, and when performing remote administration.
- NDcPP21:FCS_NTP_EXT.1: The TOE provides support for NTP v4 (RFC 5905 compliant). The connection between the TOE and the NTP server is protected via an IPsec connection.
- NDcPP21:FCS_RBG_EXT.1: The product uses an AES-256 CTR_DRBG with a software based noise source with a minimum of 256 bits of non-determinism.
- NDcPP21:FCS_TLSS_EXT.1: The TOE supports TLS v1.1 and v1.2 with the ciphersuites listed above for HTTPS and Web GUI remote administration.

6.3 User data protection

When an incoming network frame is received by the TOE, it is written by the network interface controller into kernel message buffers. Each kernel buffer is associated with a separate header that keeps track of the number of bytes of data in the buffer. The kernel clears the header prior to reading new data, and the header is updated with the count of bytes transferred by the controller.

When the buffer resource is abstracted into a message object, the object is initialized to refer only to data that has actually been overwritten in the context of the current message. This ensures that any residual information that might remain in the kernel buffer resource from previous messages is made unavailable.

State information resources that are allocated as part of the packet processing are cleared before use. This ensures that residual information that might remain from another packet is not retained.

The User data protection function is designed to satisfy the following security functional requirements:

- STFFW13:FDP_RIP.2: The TOE ensures that previous information contents of resources used for new objects are not discernible in any new object, such as network packets, as described above.

6.4 Stateful Traffic Filtering Firewall

Every IPv4/v6 packet received by the WatchGuard Firebox is intercepted by the firewall kernel. Fragmented packets are first reassembled. IPv4/v6 packets with unauthorized IP options (e.g. source route option) are dropped.

The TOE supports logical interfaces. The logical interface over which the packet was received determines the Virtual System identifier (VSID). The default VSID is 0. Each Virtual System (VS) maintains its own tables, in which only its associated (physical and logical) interfaces are registered. Each VS is allocated an independent set of processes for information flow processing within its context. An incoming packet is dispatched for processing by the corresponding Virtual System, determining the selection of the state tables and security policy that will be used to process the packet.

When an IP packet is received on a network interface, its source address is compared to topology information configured by the authorized administrator. If the source address does not correspond to the set of network addresses that match the given network interface, the packet is dropped as a spoofed packet. Note that broadcast and loopback addresses are never considered valid source addresses and are therefore rejected.

ESP-encapsulated packets are first decrypted and verified. The packet header attributes are used to match the packet against state tables that contain accepted FTP 'connections'. If the packet is successfully matched and passes packet sanity checks (correct sequence number, acknowledgment number, flags (SYN; ACK; RST; FIN.), then it is concluded that a decision has been already made for this traffic flow, and processing may skip past inspection. New ftp connections are tracked and flags in a state table are used to know when to clear the connection. The state table is cleared when the connection is closed. The TOE maintains and updates the state table to keep track of creation, open, and removal sessions. To help determine whether a packet can be part of a new session or an established session, the TOE uses information in the packet header and protocol header fields to determine the session state to which the FTP packet applies.

Immediately following the anti-spoofing verification described above, all enabled default rules are checked prior to any configured rules and dropped or allowed as appropriate.

TCP and UDP packets matching an allowed, existing session pass through the TOE's filtering rules without further inspection. When TCP or UDP packets have a source/destination address, source/destination port, and sequence number (for TCP) that matches an existing session, packets flow without inspection past the network layer until the existing session times out or the session is closed.

The TOE provides "SYN Attack" protections that allow the administrator to configure tracking of half-open TCP connections for all hosts protected by the appliance. Upon configuring a threshold number of half-open TCP connections (default of 1000), the TOE, upon detecting that number of SYN requests and the corresponding SYN-ACK responses (from a host), drops subsequent TCP SYN packets destined for the host. The TOE ages and removes half-open TCP connections based upon the administratively configured global "TCP start timeout" (default of 25 seconds). After expiring enough half-open TCP sessions, the TOE stops dropping new TCP SYN packets until the threshold is exceeded. "SYN Attack" checking is done at the time of default rules (described above).

Lastly, for all other packets, inspection is performed against the firewall rules. The rules have 4 possible outcomes:

1. Accept - the packet is allowed through;
2. Drop – the packet is dropped without notification to the sender;
3. Reject – the packet is dropped and the presumed sender is notified.
4. If no rule is matched, packets are dropped. The default drop rule is a final default rule that takes affect without administrator action. If no matching rule is configured by the administrator, all remaining packets will be dropped without a log. If the administrator wishes to log packets matching the default-deny rule, one can configure an ending drop-all rule and enable logging for it.

Firewall rules can be set to filter on protocol, source address, destination address, source port, destination port, ICMP type or ICMP code. All protocols including icmpv4, icmpv6, ipv4, ipv6, tcp, and udp may be used in configured firewall rules. Configured firewall rules are applied in the order configured by the administrator. This order can be modified at any time. If any interface is overwhelmed with traffic, it will drop the packets. An administrator can configure logging for a specific rule during configuration of the rule in the Firebox's interface.

The firewall will drop all of the following types of packets and may optionally log them if configured to do so:

1. Packets which are invalid fragments, including a description of what constitutes an invalid fragment
2. Fragments that cannot be completely re-assembled
3. Packets where the source address is equal to the address of the network interface where the network packet was received

4. Packets where the source address does not belong to the networks associated with the network interface where the network packet was received, including a description of how the TOE determines whether a source address belongs to a network associated with a given network interface
5. Packets where the source address is defined as being on a broadcast network
6. Packets where the source address is defined as being on a multicast network
7. Packets where the source address is defined as being a loopback address
8. Packets where the source or destination address of the network packet is a link-local address
9. Packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified

Further, the following types of packets should be dropped by default, if the administrator configures the TOE according to the Admin Guide. During CC compliant configuration, rules will be created to drop the following traffic as well:

1. Packets where the source or destination address of the network packet is defined as being an address 'reserved for future use' as specified in RFC 5735 for IPv4
2. Packets where the source or destination address of the network packet is defined as an 'unspecified address' or an address 'reserved for future definition and use' as specified in RFC 3513 for IPv6

During the WatchGuard Firebox boot process, there is a lag between the time when the network interface is operational, and the time that the Stateful Traffic Filtering functionality is fully functioning. During this time, Boot Security is enforced:

- Traffic flow through the appliance is disabled; and
- Traffic to and from the appliance is controlled by a Default Filter that drops all external traffic to the appliance

When the amount of traffic an interface can handle in a period of time is exceeded, any excess packets are dropped and an alert log is generated. The TOE does not allow this excess traffic to pass through the firewall uninspected and instead drops all packets, creating a log to alert the administrator which interface is affected and showing the packet count of accepted and total packets (the difference of these values being the number of packets dropped).

The Stateful Traffic Filtering Firewall function is designed to satisfy the following security functional requirements:

- STFFW13:FFW_RUL_EXT.1/2: The TOE supports all of the required protocols, which include icmpv4 (RFC 792), icmpv6 (RFC 4443), ipv4 (RFC 791), ipv6 (RFC 2460), tcp (RFC 793), and udp (RFC 768). Conformance with the RFCs defining these protocols is asserted by WatchGuard based upon WatchGuard's implementation and design. The firewall rules implement the SPD rules (permit, deny, bypass). Each rule can be configured to log status of packets pertaining to the rule. All codes under each protocol are implemented. Rules can also be assigned to each network interface. The TOE supports UDP and TCP for stateful filtering. The TOE's firewall rules apply to all IP ranges.

6.5 Identification and authentication

The TOE provides a password mechanism for authenticating users. Users are associated with a username, password, and one or more roles. Users may authenticate locally via a serial connection for the CLI or remotely via the WebUI. With 'csfc mode' enabled, SSH access to the device is disabled. Passwords can be composed of any alphabetic, numeric, and a wide range of special characters (identified in FIA_PMG_EXT.1). Passwords are not echoed back when users log on to the TOE. Internally the TOE keeps track of failed login attempts. If an administrator fails for a configured number of attempts, the administrator is either locked out for a period of time or until the primary administrator unlocks the account. The primary administrator can always log into the device via the local serial CLI connection and can never be locked out from this login.

The TOE requires identification and authentication before allowing access. The banner may be presented before authentication is complete. Once an administrator logs on to the device, administrative capabilities are unlocked.

The TOE supports X.509v3 certificates for IPsec authentication as well. X.509v3 certificates are stored internally and the store is protected by file permissions. X.509 certificates are manually loaded by the authorized administrator onto the TOE.

The authorized administrator configures the VPN peers for administrator and VPN communications, and has the option to specify a full DN, an IP address, an FQDN, or a user FQDN. The IP address, FQDN, and user FQDN fields can be used to match to a certificate's CN or SAN respective reference identifier. When an incoming request comes in, the TOE matches the peer's IP address to its configuration to find the correct rule and then attempts to match the configured reference identifier to the peer certificate. The TOE first checks the CN version of the configured reference identifier. If the verification fails, the TOE falls back to the SAN identifier and compares it. Any successful comparison will result in a valid verification while a failure of the CN then SAN results in a failed verification. The TOE then validates that it can construct a certificate path from the client's certificate through any intermediary CAs to the CA certificate specified (if it is specified) by the user in the VPN configuration. If the TOE can successfully build the certificate path, then the TOE will next check the validity of the certificates (e.g., checking its validity dates and that the CA flag is present in the basic constraints section for all CA certs). Assuming the certificates are valid, the TOE finally checks the revocation status of all certificates using OCSP. The TOE will reject any certificate for which it cannot determine the validity and rejects the connection attempt.

The Identification and authentication function is designed to satisfy the following security functional requirements:

- NDcPP21:VPNGW10:FIA_AFL.1: The TOE allows an administrator to configure a locking policy for administrative logins to the WebUI or remote CLI (the TOE does not lock out its administrative serial interface). In addition to specifying the maximum number of incorrect attempts, the administrator can specify whether the TOE should unlock the Administrator's account (after a configurable number of minutes), and if not, then the Administrator's account remains locked until another Administrator with sufficient privileges unlocks the affected account. The TOE allows an administrator to set the number of failed attempts to a value from 4-10 and to enable a lock-out time between 1-120 minutes. Again, the local access serial CLI remains available when the remote account is locked out.
- NDcPP21:FIA_PMG_EXT.1: The TOE supports passwords of varying length and allows an administrator to specify a minimum password length between 8 and 32 characters long. The password composition can contain all of the special characters required by this requirement.
- VPNGW10:FIA_PSK_EXT.1: The TOE can support a pre-shared key length of 1-79 characters for IPsec connections. Text based pre-shared keys are not conditioned and the TOE can accept bit-based pre-shared keys. The pre-shared key composition can contain all of the special characters required by this requirement. The text-based pre-shared keys are conditioned using SHA-512 hash.
- NDcPP21:FIA_UAU.7: Authentication data entered in by an administrator is obscured during login.
- NDcPP21:FIA_UAU_EXT.2: The TOE's authentication mechanism employs a locally stored database of authentication data.
- NDcPP21:FIA_UIA_EXT.1: The TOE is able to display a warning banner in accordance with FTA_TAB.1.
- NDcPP21:FIA_X509_EXT.1/Rev: OCSP revocation checking is supported for X509v3 certificate validation. Certificates are validated as part of the authentication process when they are presented to the TOE and when they are loaded into the TOE.
- NDcPP21:VPNGW10:FIA_X509_EXT.2: Certificates are checked and if found not valid are not accepted or if the OCSP server cannot be contacted for validity checks, then the certificate is not accepted. Certificates are used for validation with IPsec.
- NDcPP21/VPNGW10:FIA_X509_EXT.3: The TOE generates certificate requests and validates the CA used to sign the certificates.

6.6 Security management

User accounts are associated with roles. User accounts associated with all privileges in their role are called administrators. Authorized Administrator can access audit configuration data, firewall and VPN settings, user and administrator security attributes (including passwords), warning banner configuration, and cryptographic support settings. Further, the TOE offers a user account called a Status account. This is a read-only account that allows a user to view different status information about the current state of the TOE.

The TOE offers two administrative interfaces – command line and web-based GUI. The TOE offers command line functions which are accessible via the CLI. The CLI is a text-based interface which can only be accessed from a directly connected terminal. The CLI interface can be accessed on each evaluated device. While the CLI contains much of the base functionality needed to configure the Firebox firewall, it is recommended to use the WebUI as its available commands are all inclusive of the available CLI commands. Additionally, WebUI connections which are protected via TLS/HTTPS, can be used to administer the device remotely Typically, most authorized administrators use the Web GUI interface for management.

WatchGuard Firebox NGFWs must be directly updated, either via the WebUI or CLI.

Once authenticated, authorized administrators have access to the following security functions:

- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using digital signature and [no other] capability prior to installing those updates;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- Ability to manage the cryptographic keys;
- Ability to configure the cryptographic functionality;
- Ability to configure the lifetime for IPsec SAs;
- Ability to import X.509v3 certificates to the TOE's trust store;
- Ability to enable, disable, determine and modify the behavior of all the security functions of the TOE identified in this PP-Module;
- Ability to configure all security management functions identified in other sections of this PP-Module;
- Ability to configure audit behavior,
- Ability to modify the behavior of the transmission of audit data to an external IT entity,
- Ability to configure the reference identifier for the peer,
- Ability to configure the lifetime for IPsec SAs,
- Ability to re-enable an Administrator account,
- Ability to set the time which is used for time-stamps;
- Ability to configure NTP
- Ability to enable/disable a ruleset on a network interface
- Ability to configure a ruleset
- Ability to configure the supported algorithms.

The Security management function is designed to satisfy the following security functional requirements:

- NDcPP21:FMT_MOF.1/Functions: Only the authorized administrator has the ability to modify settings related to audit logging. Once authenticated, the administrator can modify settings related to the transmission of audit data to an external IT entity and whether or not audits are stored locally. Further, there is a "read-only" administrator account that allows one to determine the behaviour of all of the above, but this user administrator cannot modify any of the values.
- NDcPP21:FMT_MOF.1/ManualUpdate: Only the authorized administrator can update the TOE.
- NDcPP21:FMT_MTD.1/CoreData: Only the administrator can configure TSF-related functions

- NDcPP21/VPNGW10:FMT_MTD.1/CryptoKeys: Only the authorized administrator can perform operations on cryptographic keys and certificates used for the VPN.
- NDcPP21/VPNGW10:FMT_SMF.1: The TOE provides administrative interfaces to perform the functions identified above.
- STFFW13:FMT_SMF.1/FFW: The TOE provides administrative interfaces to perform the functions identified above.
- NDcPP21:FMT_SMR.2: The TOE supports administrator roles. The TOE is able to create roles for each configured administrator. An administrator can login to the TOE locally or remotely.

6.7 Packet Filtering

The packet filtering function is the VPN extended package is addressed entirely by the FFW_RUL_EXT.1 requirement. See Section 6.4.

The Packet Filtering function is designed to satisfy the following security functional requirements:

- VPNGW10:FPF_RUL_EXT.1: Please see Section 6.4 Stateful Traffic Filtering Firewall above for a description of the TOE's packet filtering capabilities.

6.8 Protection of the TSF

The TOE is an appliance and are designed to not offer general purpose operating system interfaces to users. The TOE is designed to not provide access to locally stored passwords and also, while cryptographic keys can be entered, the TOE does not disclose any cryptographic keys stored in the TOE.

The TOE components are hardware appliances that includes a real-time clock. The TOE can be configured to synchronize its clock with an NTP time server. The TOE uses the clock to support several security functions including timestamps for audit records, timing elements of cryptographic functions, and inactivity timeouts. Furthermore, to protect the communication between the TOE and the NTP server, an IPsec tunnel can be used to protect communications for NTP (similar to syslog). The TOE also supports timekeeping without an NTP server, in the event that no NTP server is configured or connection to the server is lost.

During power-up the integrity of all executables is verified with a digital signature. The public key used for signature verification comes pre-installed on the TOE. If an integrity test fails in the cryptographic module, the system will halt boot and display a failure warning, requiring a power cycle to recover.

During power-up, cryptographic algorithms functionality is tested by taking a series of fixed inputs, running them through the cryptographic module, and comparing the output to saved, known-answer responses. If these self-tests fail during boot, a boot log is generated stating that a self-test failure occurred, and the system boot is halted. A power cycle is required to restart the device.

Additionally, noise source health tests are continually executed during runtime to verify that the noise source continues to operate at a high quality.

The TOE performed the following power-up cryptographic algorithm known answer tests:

Algorithm	Implemented in	Description
AES encryption/decryption	Crypto Module	Comparison of known answer to calculated value
DRBG random bit generation	Crypto Module	Comparison of known answer to calculated value
ECDSA sign/verify	Crypto Module	Comparison of known answer to calculated value
HMAC-SHA	Crypto Module	Comparison of known answer to calculated value
RSA sign/verify	Crypto Module	Comparison of known answer to calculated value
SHA hashing	Crypto Module	Comparison of known answer to calculated value
KAS (ECC/FCC)	Crypto Module	Comparison of known answer to calculated value

Table 6 Power-up Cryptographic Algorithm Known Answer Tests

The TOE supports loading updates by the administrator using either management interface. The administrator must manually obtain the update from the WatchGuard web site (https://watchguardsupport.secure.force.com/software/). The TOE automatically verifies the image's digital signature during install. If an image's digital signature verification fails, the update is halted, the loaded image is discarded, and the TOE continues operating uninterrupted.

Common criteria evaluated images and updates for the T35/T55/T70 are made available on request. Administrators can obtain T35/T55/T70 images by contacting WatchGuard support.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

- NDcPP21:FPT_APW_EXT.1: The TOE does not offer any functions that will disclose to any user a plain text password. Furthermore, locally defined passwords are not stored in plaintext form.
- VPNGW10:FPT_FLS.1/SelfTest: The TOE contains known-answer self-tests that are executed during power-up. The TOE enters an error state and does not boot when it fails its self-tests. The cryptographic module is signed with a digital signature using ECDSA P-521 and SHA-512.
- NDcPP21:FPT_SKP_EXT.1: The TOE does not provide any means for reading any key or CSP.
- NDcPP21:FPT_STM_EXT.1: The TOE provides reliable time stamps using an internal clock, and can also synchronize its clock with an external NTP server (through an IPsec tunnel).
- NDcPP21/VPNGW10:FPT_TST_EXT.1: The TOE runs known-answer self-tests on the Cryptographic Module and image integrity verification during power-up along with continual run-time noise source health tests.
- VPNGW10:FPT_TST_EXT.3: During power-up the integrity of all executables is verified with a digital signature.
- NDcPP21:VPNGW10:FPT_TUD_EXT.1: The TOE offers an interface to query the current version of itself. The TOE must be updated manually through a manual download and update. The TOE's updates are digitally signed and verified using ECDSA with SHA-512 and P-521 curve. Upon a failed verification, the update is halted and the TOE continues operation uninterrupted.

6.9 TOE access

The TOE can be configured to display an administrator-configured message of the day banner that will be displayed before authentication is completed. The banner will be displayed when accessing the TOE via the console or web interfaces.

The TOE provides an inactivity timeout for both WebUI and serial/remote CLI connection sessions. The authorized administrator can set the inactivity timeout for each interface; this timeout is separated by the UI type: local or remote CLI and remote WebUI. When an inactivity period is exceeded, the session is terminated. The user will be required to login in after any session has been terminated due to inactivity or after voluntary termination. To prevent total lockout, the main/default admin always has the ability to login via a serial CLI connection to the TOE.

The TOE provides an interface to configure restrictions on which VPN clients can connect to the TOE. An administrator can deny VPN connections from clients based on location (IP Address), time, and day.

The TOE access function is designed to satisfy the following security functional requirements:

- NDcPP21:FTA_SSL.3: The TOE allows remote inactive sessions to disconnect after a set period of time configurable in the WebUI.
- NDcPP21:FTA_SSL.4: The TOE allows session disconnect via a logout command.
- NDcPP21:FTA_SSL_EXT.1: The TOE is able to lock a local administrator session after a set inactivity time, requiring re-authentication before TSF functionality is made available.

- NDcPP21:FTA_TAB.1: The TOE supports a message of the day banner that is displayed when an administrator authenticates to the TOE both locally and remotely.

6.10 Trusted path/channels

The TOE uses TLS/HTTPS and IPsec to protect communications. The TOE employs IPsec when it sends audit data to an audit server, when communicating with an NTP server, and when allowing remote CLI administration connections. The TOE employs TLS/HTTPS connections when an administrator uses the Web GUI to administer the TOE. Authorized local administrators can only connect to the TOE directly via a serial connection. While the serial connection is not protected, it requires local, direct access to the individual device. In all remote cases, IPsec or TLS/HTTPS ensures traffic is not modified or disclosed. The nature of local connections ensures that interfacing with the CLI via a serial connection is secure.

The Trusted path/channels function is designed to satisfy the following security functional requirements:

- NDcPP21:FTP_ITC.1: The TOE uses IPsec to provide a trusted communication channel between itself and an audit and/or NTP server. Additionally, IPsec can be used for VPN communications. HTTPS/TLS is used for secure remote administration of the TOE.
- VPNGW10:FTP_ITC.1/VPN: The TOE can use IPsec to provide protected communication between the TOE and IT entities. Further, the TOE can initiate the trusted channel communication for remote VPN peers.
- NDcPP21:FTP_TRP.1/Admin: The TOE implements IPsec and 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.

7. Hardware Platforms

Below is a list of hardware platforms included in the evaluation. Each of the hardware platforms runs the same version of software. Each of the evaluated platforms are separated into groups of equivalence based on their chipset. Each individual row represents equivalent processor family devices.

Software Version	Hardware Models (bold = tested, grey = equivalent)	CPU/CPU Family
FirewareOS 12.6.2 Linux Kernel 4.14.83	**T35**	NXP T1024 (PPC) – e500
	T40 *T20*	NXP Layerscape LS1023A, LS1043A – ARM Coretex-A53
	T80	NXP Layerscape LS1046A – ARM Coretex-A72
	T55 *T70*	Intel Celeron N3060, N3160 (x86) – Braswell
	M270	Intel Atom C3558 (x86) – Denverton
	M370	Intel Celeron G3900 (x86) – Skylake
	M470	Intel Pentium G4400 (x86) – Skylake
	M570	Intel Core i3-6100 (x86) – Skylake
	M670	Intel Skylake E3-1225v5 (x86) – Haswell
	M4600	Intel Xeon E3-1275V3 (x86) – Haswell
	M5600	Intel Xeon E5-2680V2 (x86) – Ivy Bridge

Table 7 TOE Hardware Platforms and Details

Below is the Ethernet Controller/Driver breakdown. The color coding represents controller/driver equivalence. The **bold** platforms represent the platforms that were tested during the firewall portion of the evaluation. Note: testing was repeated for both the T55 and M4600 devices (despite equivalence) to ensure no different behavior was detected due to the Marvell DSA switch wired into the device's Ethernet Controller. Testing proved that the Marvell DSA acted only to forward traffic and did not modify behavior of the firewall functionality.

Note: Intel I210-IS is equivalent to the Intel I210-AT but includes the Marvell controller – their architecture is identical otherwise.

Hardware Models (bold = tested, grey = equivalent)	Ethernet Controller/Driver
T35 **T40** *T20*	NXP DPAA
T80	NXP DPAA w/ Marvell DSA

Hardware Models (bold = tested, grey = equivalent)	**Ethernet Controller/Driver**
T55	Intel I210-IS igb w/ Marvell DSA
T70	Intel I210-AT igb and Intel I210-IS igb w/ Marvell DSA
M4600 *M270* *M370* *M470* *M570* *M670* *M5600*	Intel I210-AT igb
8x1g Expansion Module (WG8592 and WG8593)	Intel I350-AM4 ixgbe
4x10g Expansion Module (WG8594)	Intel 82599ES ixgbe
2x40g Expansion Module (WG8023)	Intel XL710 i40e

Table 8 TOE Ethernet Port/Driver Details